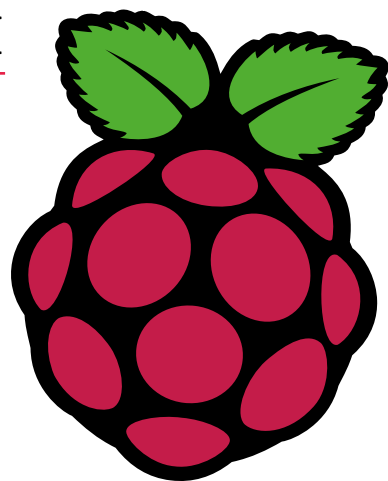




BUY IN PRINT **WORLDWIDE** MAGPI.CC/STORE

The MagPi



Issue 96

August 2020

magpi.cc

The official Raspberry Pi magazine

25

NO-PROBLEM PROJECTS

Go ahead! Make something
with your Raspberry Pi today

Robot car test!



+ 10 summer projects

Run Windows 98 software

**Keep out! With a calendar
controlled busy light**



**Build a home
music system**

 **GLOBAL
DELIVERY**
magpi.cc/store



46 PAGES OF PROJECTS & TUTORIALS

Delivering Excellence in 48 hours to most areas in Europe

**FREE
SHIPPING**
ON ORDERS OVER
£33 OR \$50 USD



0800 587 0991
DIGIKEY.CO.UK



9.2 MILLION+ PRODUCTS ONLINE | 1,100+ INDUSTRY-LEADING SUPPLIERS | 100% FRANCHISED DISTRIBUTOR

*A shipping charge of £12.00 will be billed on all orders of less than £33.00. A shipping charge of \$18.00 USD will be billed on all orders of less than \$50.00 USD. All orders are shipped via UPS, Federal Express, or DHL for delivery within 1-3 days (dependent on final destination). No handling fees. All prices are in British pound sterling or United States dollar. Digi-Key is a franchised distributor for all supplier partners. New products added daily. Digi-Key and Digi-Key Electronics are registered trademarks of Digi-Key Electronics in the U.S. and other countries. © 2020 Digi-Key Electronics, 701 Brooks Ave. South, Thief River Falls, MN 56701, USA

ECIA MEMBER
Supporting The Authorized Channel

WELCOME to *The MagPi* 96

We all started somewhere. I remember my first electronics project, a small electronics kit I bought from Maplin (sadly, no longer on the high street). After that, I built a CamJam EduKit #3 robot and then there was no stopping me.

The hurdle between setting up a Raspberry Pi and making something trips up most of us. Raspberry Pi newbies need ideas, a bit of inspiration, and just a smidgen of encouragement. Building something cool with Raspberry Pi is easier than most newcomers imagine.

And we want them to discover the joy of making with Raspberry Pi. This month's feature is 25 no-nonsense projects (Page 34). Inside, it has the best entry-level projects, kits, and ideas for incredible things to make. All of them are way easier than you'd think.

We want as many people as possible to discover the joy of Raspberry Pi computing. That's why this month Rob has taken a look at Accessibility in Making. Discover how to make projects that work for all people (Page 70).

And, of course, we've got a whole issue packed with the most amazing projects and superb tutorials. There's something for everyone here.

Lucy Hattersley Editor



EDITOR

Lucy Hattersley

Lucy is editor of *The MagPi* and this month she's been making her fourth "wheely great" robot. She's considering starting a robot convoy.

@LucyHattersley



GET A
**RASPBERRY
ZERO W KIT**
WITH A SUBSCRIPTION!
PAGE 32



The
MagPi

HackSpace
TECHNOLOGY IN YOUR HANDS

CustomPC

3 ISSUES FOR £10



FREE BOOK



magpi.cc/freebook

Subscribe to The MagPi, HackSpace magazine, or Custom PC. Your first three issues for £10, then our great value rolling subscription afterwards. Includes a free voucher for one of five fantastic books at store.rpiexpress.cc/collections/latest-bookazines
UK only. Free delivery on everything.

Contents

► Issue 96 ► August 2020

Cover Feature

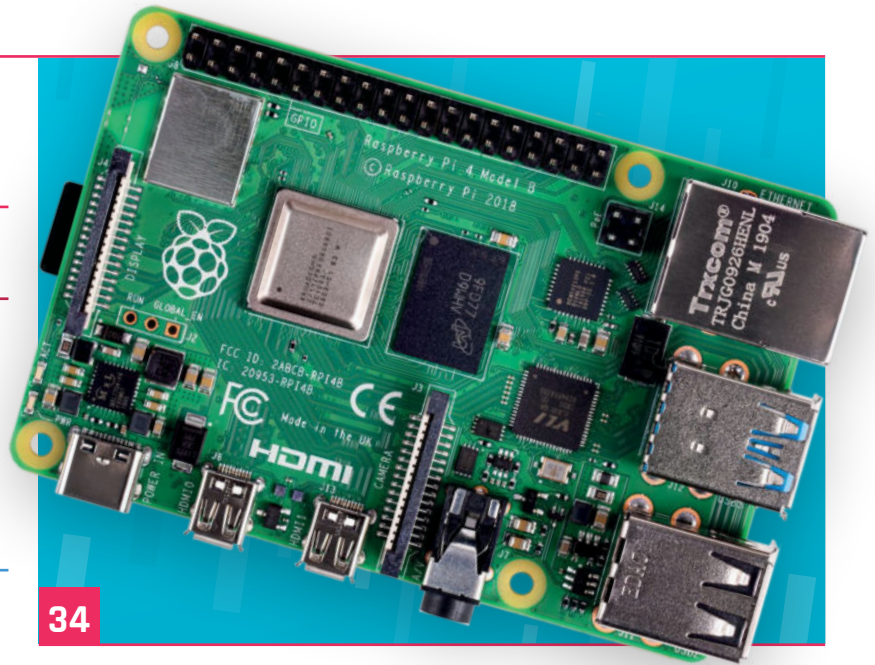
34 25 no-problem projects

Regulars

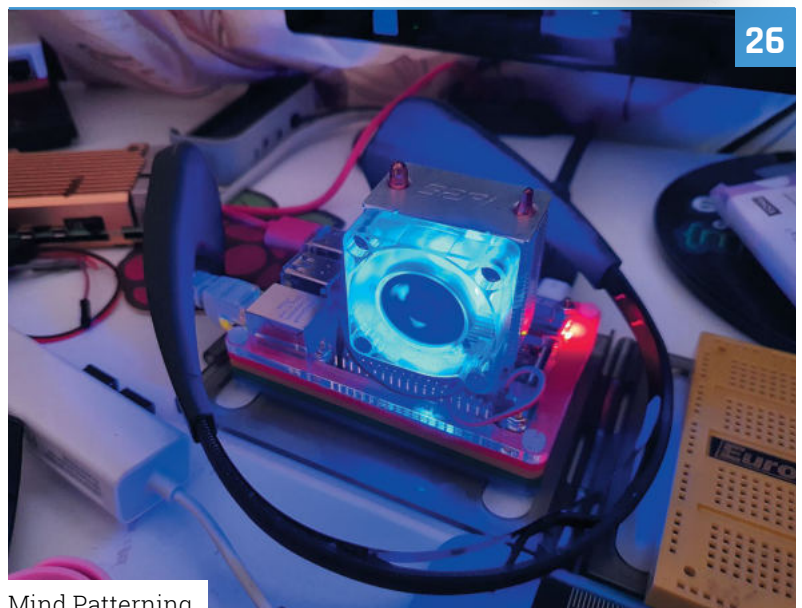
- 08 World of Raspberry Pi
- 92 Your Letters
- 97 Next Month
- 98 The Final Word

Project Showcases

- 10 Project Zed
- 14 Sonos / Spotify Vinyl Emulator
- 16 ISS Tracker
- 18 PiMowBot
- 22 OpenFlexure Microscope
- 24 Calendar-Controlled Busy Light
- 26 Mind Patterning
- 28 LEDWeatherBoard



34



26



18

PiMowBot

Mind Patterning

DISCLAIMER: Some of the tools and techniques shown in The MagPi magazine are dangerous unless used with skill, experience, and appropriate personal protection equipment. While we attempt to guide the reader, ultimately you are responsible for your own safety and understanding the limits of yourself and your equipment. Children should be supervised. Raspberry Pi (Trading) Ltd does not accept responsibility for any injuries, damage to equipment, or costs incurred from projects, tutorials or suggestions in The MagPi magazine. Laws and regulations covering many of the topics in The MagPi magazine are different between countries, and are always subject to change. You are responsible for understanding the requirements in your jurisdiction and ensuring that you comply with them. Some manufacturers place limits on the use of their hardware which some projects or suggestions in The MagPi magazine may go beyond. It is your responsibility to understand the manufacturer's limits.

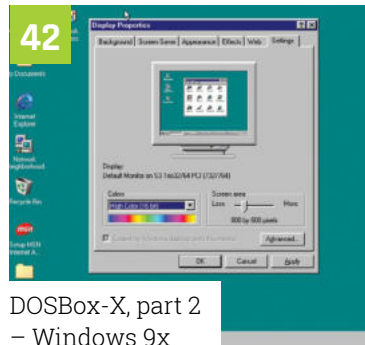
Tutorials

- 42** DOSBox-X, part 2 – Windows 9x
- 48** HQ Camera: stop-motion
- 52** Home Music System, part 1
- 56** Remote access a Raspberry Pi
- 62** Clock-controlled devices
- 66** Supercharged Scope – Pi Bakery

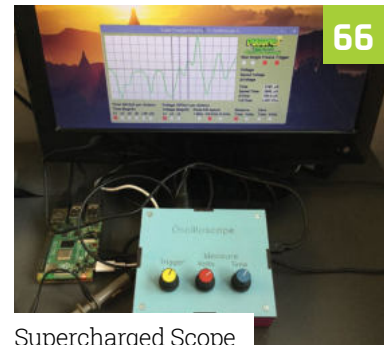
The Big Feature



Make making accessible



DOSBox-X, part 2
– Windows 9x



Supercharged Scope



PiCar-V Kit V2.0

Reviews

- 76** PiCar-V Kit V2.0
- 78** UCS Universal Housing
- 79** Ren'py
- 80** 10 Best: Summer projects
- 82** Learn C

Community

- 84** Masafumi Ohta interview
- 86** This month in Raspberry Pi
- 90** Events calendar



Masafumi Ohta interview

WIN
1 OF 10

ARGON NEO CASES

95

PHILIPS

Monitors

Brilliant interactive display with Smooth Touch

innovation ✨ you



B line touch monitor



Water and Dust
Resistant



SmartContrast



LowBlue Mode

Available at: **Softcat** **Insight**





Raspberry Pi 4 web hosting with Mythic Beasts

Web service slashes costs and doubles speeds with Raspberry Pi 4 server space. By **Rosie Hattersley**

▲ A cluster of Raspberry Pi 4 computers are used to serve millions of pages on the Raspberry Pi blog

Mythic Beasts is an ISP (internet service provider) that offers a Raspberry Pi-based cloud for web developers.

Recently, Mythic Beasts added Raspberry Pi 4 boards to its cloud. These low-cost computers are big hitters in the server world, outperforming many other rivals in the web hosting arena, while also undercutting them.

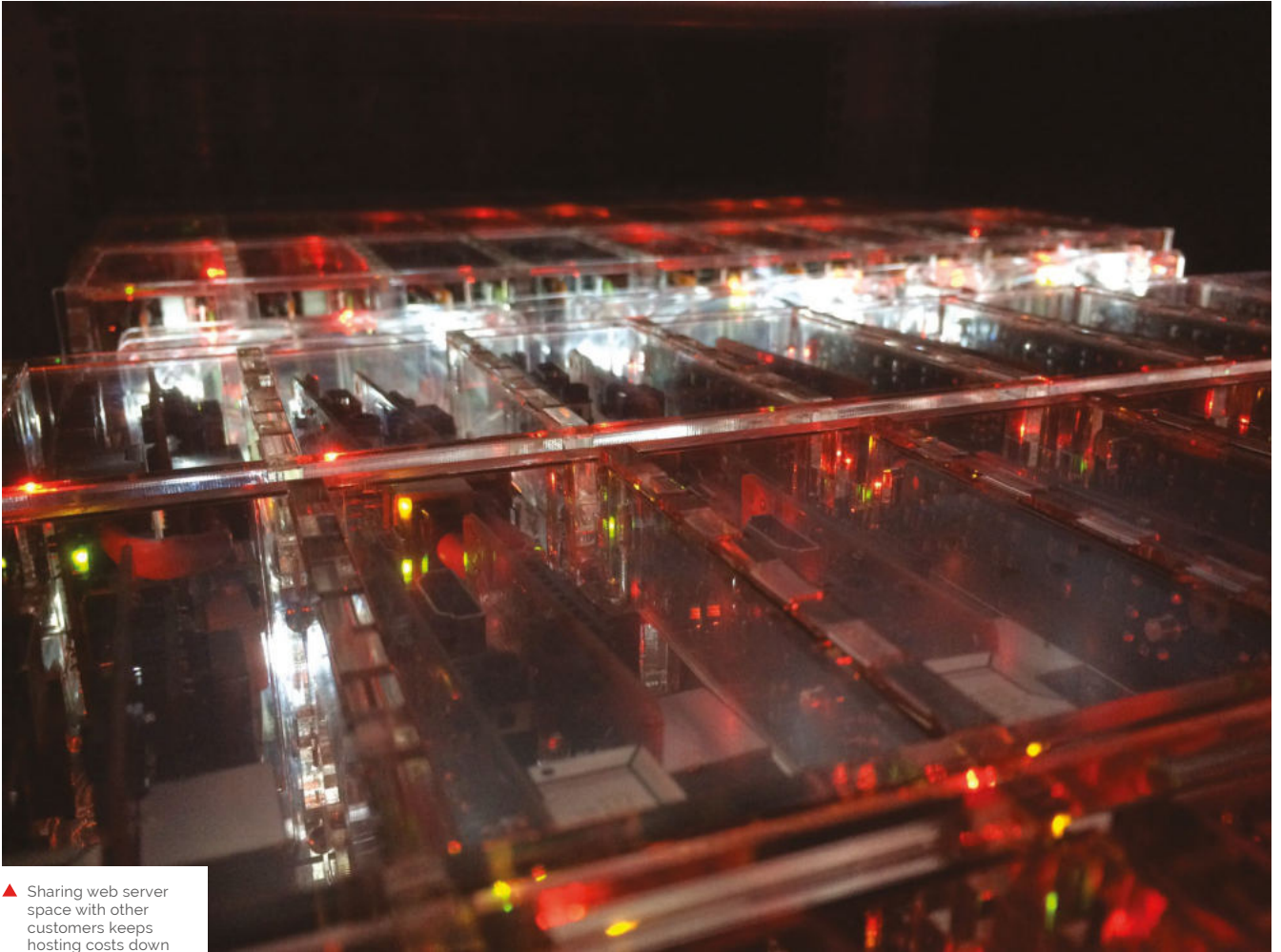
“Mythic Beast's testing compares favourably against rivals such as AWS (Amazon Web Services)”

“It's taken us a little while to make it happen,” writes Pete Stevens, Director at Mythic Beasts. “Testing started prior to Raspberry Pi 4's launch in June 2019, after Eben [Upton] dropped 18 [4GB] Raspberry Pi 4 computers round.”

Despite the tens of millions of site visits on launch day, Raspberry Pi 4 in the cloud performed flawlessly. “The launch was almost perfectly smooth,” says Pete.

Rock-solid service


The launch day cluster test was kept on for a whole month; more than long enough to prove the hardware's mettle as a hosting platform. Mythic Beasts is now offering Raspberry Pi 4 devices in its cloud, starting from £7.50/month



▲ Sharing web server space with other customers keeps hosting costs down

or 1.2p/hour (magpi.cc/mythicorder). According to Mythic Beasts, “Raspberry Pi 4 represents a significant upgrade over Raspberry Pi 3; it is over twice as fast, has four times the RAM, and the network card runs at full gigabit speed. On a network-booted server, this gives you much faster file access in addition to more bandwidth out to the internet.”

Mythic Beast’s testing compares favourably against rivals such as AWS (Amazon Web Services), running twice as fast for a quarter of the price.

If you are interested in setting up a website using Raspberry Pi servers, take a look at the walkthrough on Mythic Beast’s website: magpi.cc/mythichosting. 

	Raspberry Pi 4	a1.large	m6g.medium
Spec	4 cores @ 1.5GHz 4GB RAM	2 cores 4GB RAM	1 core 4GB RAM
Monthly price	£8.63	\$45.35 (~£36.09)	\$34.69 (~£27.61)
Requests per second	107	52	57
Mean request time	457 ms	978 ms	868 ms
99th percentile request time	791 ms	1247 ms	1056 ms

Source: Mythic Beasts

Project Zed

A sci-fi inspired robot made from old printer parts is helping fight the good fight, just like the character that inspired it. **Rosie Hattersley** explains how



Joseph Uding

Joseph Uding's robot-building education started with Stan Winston's Hollywood animatronics and MIT's robotics online courses. He's now a project officer at Hive Colab's department of Robotics and AI.

hivecolab.org

► The robot can move around, telling people to put on their masks and to keep their distance




Joseph Uding first fell for the charms of Raspberry Pi when he was studying robotics with MIT online and was pondering the amount of computing power required for ASIMO, Honda's famous humanoid robot. "The bigger the robot, the bigger the motors and batteries, plus many things that can go wrong because a robot is made up of many computers in it," he reasons. "That's when I started loving Raspberry Pi."

The single-board computer features prominently in Joseph's Project Zed robot, created as a means of encouraging citizens to take sensible health precautions during the coronavirus pandemic. "A friend said 'build a robot that can help in fighting Ebola in West Africa'. That was in 2016 and we didn't take it seriously because no cases were in Uganda, but when 2020 came I got to work. With a robot, he says, "you can carry out random tests and collect information from someone without risking important medical workers."

Raspberry Pi enabled Joseph to add personality and voice-based interaction to the robots he's built, as well as delivering preset messages such as "don't forget to wear a face mask".

Parts and labour

Project Zed is, by necessity, a masterclass in upcycling: 3D printers and next-day component deliveries aren't reliably available in Uganda, but Joseph believes that if you can have a clear idea of what you want to make, you can make it from almost anything. "The first thing is to see the



Two cameras connected to Raspberry Pi act as the eyes

There's a speaker for a mouth and microphone for listening to commands and understanding people

Quick FACTS

- ▶ Sci-fi film characters often inspire Joseph's builds
- ▶ Cats' serious expressions are another influence
- ▶ His robots sometimes seem deep in serious chat
- ▶ He loves his Raspberry Pi 3B+ and says...
- ▶ "I can program it to do magic when it comes to robotics"

The rugged wheels of the RC car mean the robot can be used in all sorts of terrain



▲ The robot has four 12V DC motors and a solar panel to recharge the battery if it is out "doing its job in remote areas"



▶ Joseph's first ever robot, TIM, made from a TV antenna, with facial expressions by Google Assistant on an old mobile phone

problem you want to solve with the robot you want to make." His project took him just three weeks.

After work one day, he went to visit some guys who fix printers and asked them for old dead ones. "They smiled and gave me about four spoiled old printers. From there I got DC motors and plastic parts to use when housing the robot." The motors would be used to replace those in a broken RC car that would act as the robot's base and help stabilise it.

“ It can move its arms up and down, but not the fingers yet ”

The initial designs were in lightweight wood he sourced himself – “just like the way planes are made” – which was light and easy to work with. Plastic from the old printers was used for the casing.

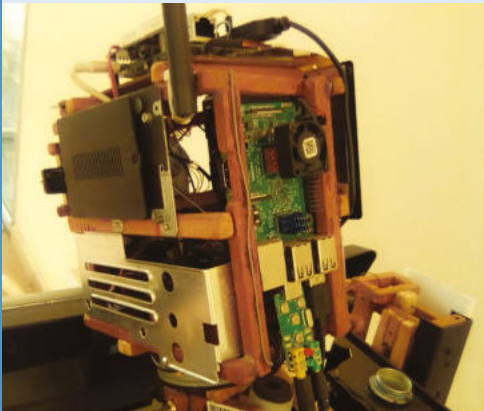
Aside from the Raspberry Pi to control things, the half-humanoid, half-RC car robot has a Roboclaw 2 motor controller (magpi.cc/roboclaw) and a Google Coral AI USB accelerator (magpi.cc/coralusbaccelerator).

It can be controlled from a distance of up to roughly 400 metres and can listen, talk, and see objects. "It can move its arms up and down, but not the fingers yet," says Joseph. "It's limited to doing one thing at once – bending down to pick

Cyber setup



- 01** Assemble parts including an RC motor and create a lightweight wooden or plastic frame. Start with parts for the head to the waist. Join together using glue and screws.



- 02** Fit the robot frame into the RC car, then install Raspberry Pi, webcam, radio speakers, microphones, and Android Things in the robot's head.



- 03** Program the robot to respond to verbal requests, such as tuning in to a specific TV station or finding a recipe. Complete the wiring and robot casing.



something up or moving itself along – because it has only one Raspberry Pi for the head where the AI is based.”

▲ Joseph upcycled an old remote-control car for his robot's base

Software trials

Joseph first tried using the open-source TensorFlow (magpi.cc/tensorflow) and YOLO for object detection, but really had in mind something more like a Movidius Neural Compute Stick to handle face detection (which he didn't have). Instead, having consulted projects such as the Personal Assistant Robot (magpi.cc/parobot), he turned to SSH and live video streaming, which did a great job. “I can control the robot from anywhere around the world using my cell phone or PC any time I want to,” Joseph explains – something with clear potential in war or disaster zones where medical workers' lives would otherwise be at risk.

“Doing such projects where you don't have the actual resources helps you to be good at recycling things and also teaches you how to be super-creative in life,” he says, modestly. [M](#)

Sonos / Spotify Vinyl Emulator

Experience the tactile nature of a vinyl music collection without the cost. **Phil King** gets all nostalgic



Mark Hankinson

Mark lives in London and enjoys running, skiing, and cartography. This is his first Raspberry Pi project, although this doesn't explain why he now owns four.

hackster.io/mark-hank

There's nothing quite like the experience of handling an old-fashioned 12-inch vinyl album, but most people don't own a turntable to play one on. Vinyl prices have also rocketed in recent years, making collecting them an expensive hobby.

For his very first Raspberry Pi project, Mark Hankinson has created a solution. "I listen to a lot of music through streaming services," he tells us, "but I never really felt like I had a 'collection' – it was a stream of tracks chosen for me by an algorithm, rather than me making real choices. I can't remember the last time I listened to an album all the way through. Like a lot of people, that led me to the idea of building a vinyl collection, but the idea of buying something I could already listen to through streaming seemed wasteful."

Aiming to replicate the 'collection' aspects and tactile, screen-free nature of vinyl, Mark came up with an intriguing solution involving the use of cardboard album art with NFC tags. When placed on a box containing an NFC reader connected

to a Raspberry Pi, the card triggers playback of the relevant music via Spotify on a Sonos sound system.

Each NFC tag can be easily programmed with an identifier for a Spotify playlist, an Apple Music album, or a TuneIn radio station, etc. "My script loops endlessly waiting until it gets an NFC input, then works out what kind of input that is and passes it to the Sonos API to do something with."

Album art

Rather than using full-size album sleeves, Mark elected to use mini versions. "I've got about 100, which I got printed for £25 and look great. I decided to go with 4-inch square photo cards as they're easy to handle, but really there's no limit to what you can do because the tags are so tiny and inexpensive."

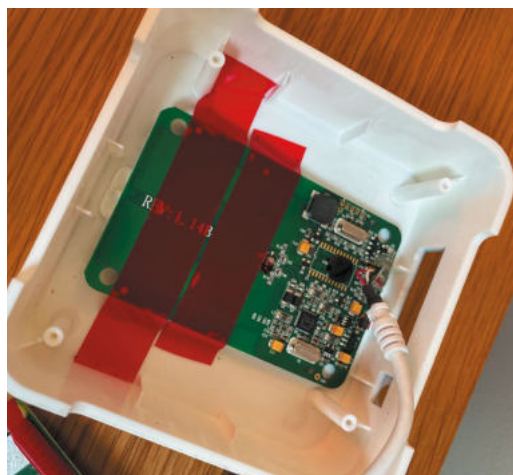
For that extra touch of nostalgia, he even has NFC tags for his favourite playlists stuck to old cassette tape boxes so that they resemble eighties-style mix tapes!

An impressive debut Raspberry Pi project for Mark, it took him just a few days to get the whole system up and running. "Luckily, a lot of the building blocks (a Python library to interface with NFC and an API to interface with Sonos) already existed, so really all I was doing was wrapping them together," he says, modestly. "There are also some similar projects created several years ago to do things through a Raspberry Pi with RFID: this is an older technology than NFC, a lot bulkier and not as flexible, but they were still really useful to see how others have approached the issues."

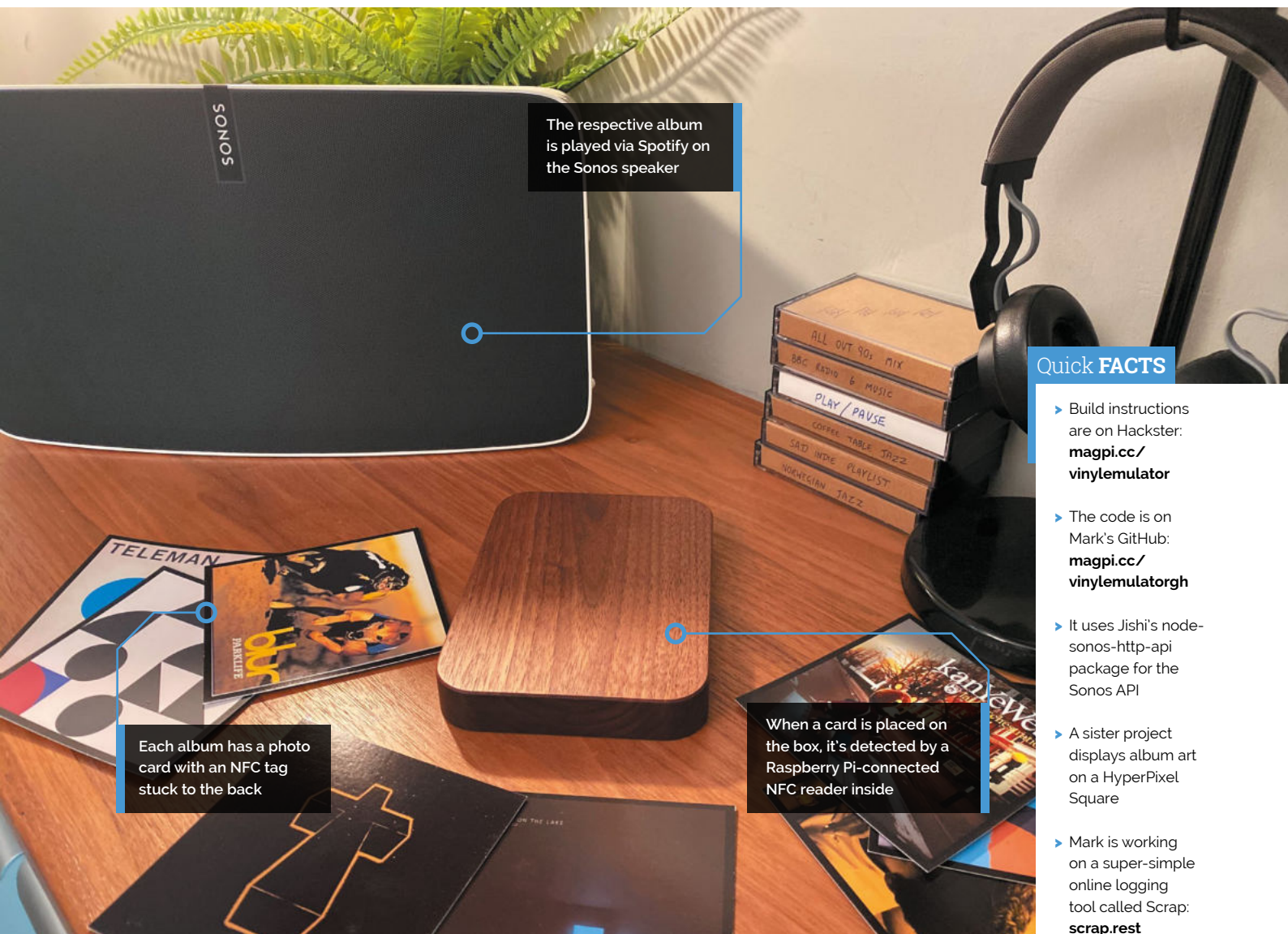
System modification

Mark says the system could easily be adapted to use a sound system other than Sonos. "It just depends on whether there is an API available for whichever speaker setup you have running. The

▼ Each of Mark's mix tapes has an NFC tag to trigger the playback of its playlist



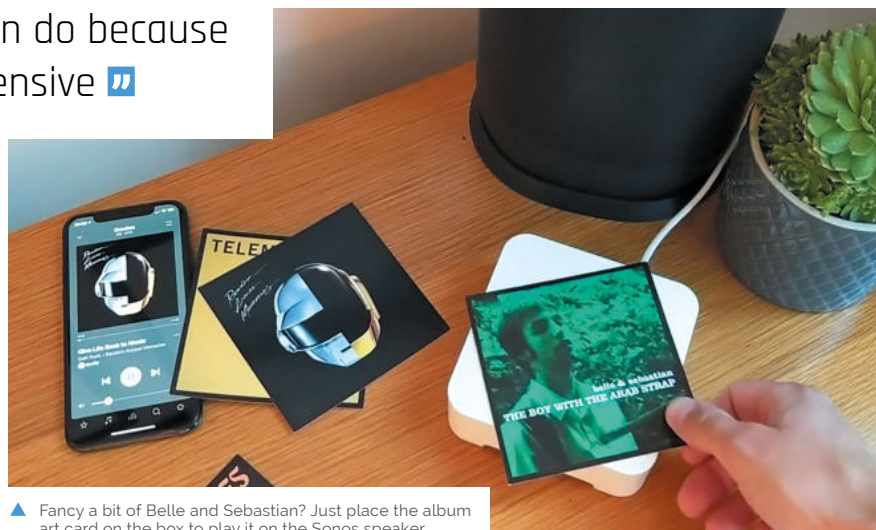
▲ A Raspberry Pi is secreted in the box along with an NFC reader



“There's no limit to what you can do because the tags are so tiny and inexpensive”

PyChromecast library looks pretty helpful for Google, for instance. But that will all be a project for someone else to build! I would be delighted to incorporate their code into the GitHub repo if someone does build it.”

Mark has set up a forum on reddit (r/vinylemulator, magpi.cc/vereddit) for people to post their implementations, and says it's great to see what some people have done with this code. “One person hid the tags inside Disney toys so that his daughter can play her favourite music from different films. Another built a mini record player out of Lego to hide the reader, and put the tags on tiny Lego discs.” [M](#)



International Space Station Tracker

Fancy tracking the ISS's trajectory? All you need is a Raspberry Pi, an e-paper display, an enclosure, and a little Python code. **Nicola King** looks to the skies



Sridhar Rajagopal

Software engineer turned hardware-hacking enthusiast and entrepreneur, Sridhar Rajagopal is the founder of Upbeat Labs and creator of ProtoStax – a maker-friendly stackable, modular, and extensible enclosure system.

magpi.cc/srajagopal

Standing on his balcony one sunny evening, the perfect conditions enabled California-based astronomy enthusiast Sridhar

Rajagopal to spot the International Space Station speeding by, and the seeds of an idea were duly sown. Having worked on several projects using tri-colour e-paper (aka e-ink) displays, which he likes for their “aesthetics and low-to-no-power consumption”, he thought that developing a way of tracking the ISS using such a display would be a perfect project to undertake.

“After a bit of searching, I was able to find an open API to get the ISS location at any given point in time,” explains Sridhar. I also knew I wouldn’t have to worry about the data changing several times per second or even per minute. Even though the ISS is wicked fast (16 orbits in a day!), this would still be well within the refresh capabilities of the e-paper display.”

Station location

His ISS Tracker works by obtaining the ISS location from the Open Notify API every 30 seconds. It appends this data point to a list, so older data is available. “I don’t currently log the data to file, but it would be very easy to add this functionality,” says Sridhar. “Once I have appended the data to the list, I call the drawISS method of my Display class with the positions array, to render the world map and ISS trajectory and current location. The world map gets rendered to one PIL image, and the ISS location and trajectory get rendered to another PIL image.”

Each latitude/longitude position is mapped to the corresponding XY co-ordinate. The last position in the array (the latest position) gets rendered as the ISS icon to show its current position. “Every 30th data point gets rendered as a rectangle, and every other data point gets rendered as a tiny circle,” adds Sridhar.

From there, the images are then simply passed into the e-paper library’s display method; one image is rendered in black, and the other image in red.

Track... star

Little wonder that the response received from friends, family, and the wider maker community has been extremely positive, as Sridhar shares: “The first feedback was from my non-techie wife who love-love-loved the idea of displaying the ISS location and trajectory on the e-paper display. She gave valuable input on the aesthetics of the data visualisation.”

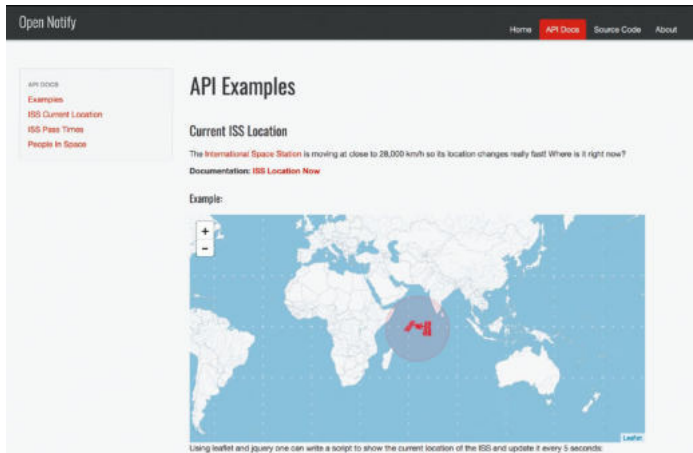
“ Every 30th data point gets rendered as a rectangle, and every other data point gets rendered as a tiny circle ”

In addition, he tells us that other makers have contributed suggestions for improvements. “JP, a Hackster community user [...] added information to make the Python code a service and have it launch on bootup. I had him contribute his changes to my GitHub repository – I was thrilled about the community involvement!”

Housed in a versatile, transparent ProtoStax enclosure designed by Sridhar, the end result is an elegant way of showing the current position and trajectory of the ISS as it hurtles around the Earth at 7.6 km/s. Why not have a go at making your own display so you know when to look out for the space station whizzing across the night sky? It really is an awesome sight. [🔗](#)



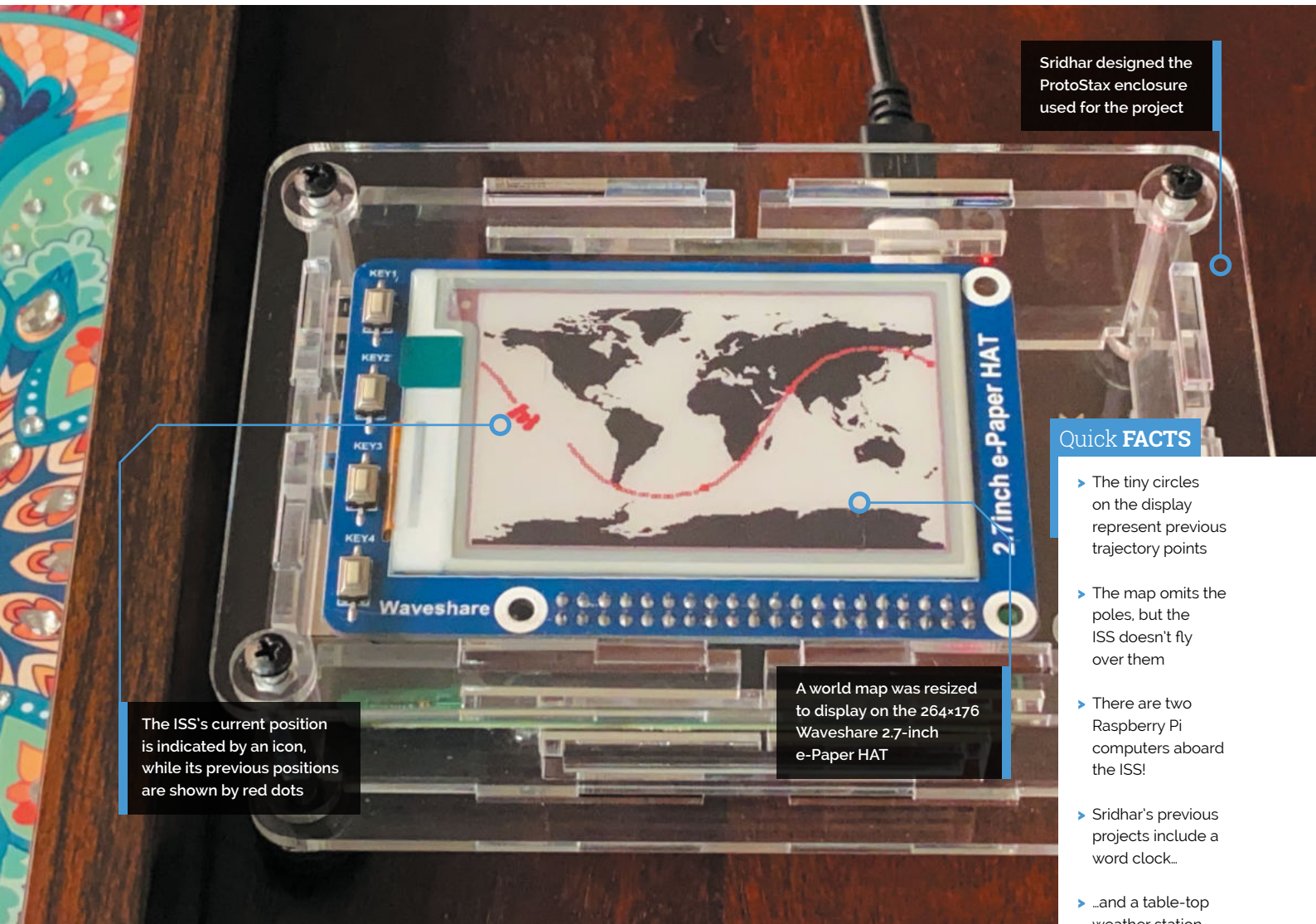
▲ The e-paper display mid-refresh. It takes about three seconds to refresh, but it’s fast enough for this kind of project



- ▲ The ISS location data is obtained using the Open Notify API – visit magpi.cc/isslocation to see its current position

```
19: import sys
20: sys.path.append(r'lib')
21:
22: if sys.version_info[0] < 3:
23:     raise Exception("Must be using Python 3")
24:
25: from enum import Enum
26: import signal
27: import apizero
28: import apiconfig
29:
30: from PIL import Image, ImageDraw, ImageFont, ImageOps
31: from datetime import datetime
32: from time import sleep
33:
34: import requests
35:
36: # Update Interval
37: INTERVAL = 30 #seconds
38:
39: # Note:
40: # The dimensions of the 2.7 inch e-paper display are
41: # 264 x 176
42:
43: class Display(object):
44:     def __init__(self, imageWidth, imageHeight):
45:         self.imageWidth = imageWidth
46:         self.imageHeight = imageHeight
47:
48:     # Draws the ISS current location and trajectory from array of positions
49:     def drawISS(self, positions):
50:         imageBack = Image.new('1', (self.imageWidth, self.imageHeight), 255) # 1: clear the frame
51:         imageMap = Image.open('world_map_m.bmp').convert('L')
52:         imageBack.paste(imageMap, (0,0))
53:
```

- ▲ The project code is written in Python and can be found on Sridhar's GitHub page: magpi.cc/isstrackercode



Sridhar designed the ProtoStack enclosure used for the project

Quick FACTS

- ▶ The tiny circles on the display represent previous trajectory points
- ▶ The map omits the poles, but the ISS doesn't fly over them
- ▶ There are two Raspberry Pi computers aboard the ISS!
- ▶ Sridhar's previous projects include a word clock...
- ▶ ...and a table-top weather station

The ISS's current position is indicated by an icon, while its previous positions are shown by red dots

A world map was resized to display on the 264x176 Waveshare 2.7-inch e-Paper HAT

PiMowBot

A pristine lawn is really something, but maintaining it can be time-consuming – unless you have a Raspberry Pi-powered robotic lawn-mower.

Rosie Hattersley learns more



Dirk Weyand

Communications technology specialist and keen DIY maker, Dirk got his first Raspberry Pi back in 2012 and loves its ability to solve real-life problems.

magpi.cc/pimowbot

Manicured lawns have obsessed homeowners for centuries, giving the illusion of scale as well as order.

However, as our lives become ever busier, maintaining a weed-free, neatly trimmed patch of grass becomes a tall order. Getting a robot lawn-mower to take the strain seems like a sensible solution, especially if, like PiMowBot creator Dirk Weyand, you have a 600 m² lawn to keep you busy.

Dirk embarked on his PiMowBot project two summers ago, aiming to create a robot that was powerful and reliable enough to mow his sizeable garden weekly. Although other robot mowers existed, such as the ArduMower (magpi.cc/ardumower), they didn't fit Dirk's strict lawn-mowing brief in which "the PiMowBot mows the lawn lane by lane so that the lawn gets an even cut." Instead, he set about creating his own robot mower, recruiting his business customer and co-designer Tim Esser



▲ A prototype version of PiMowBot shows off its generous solar panel



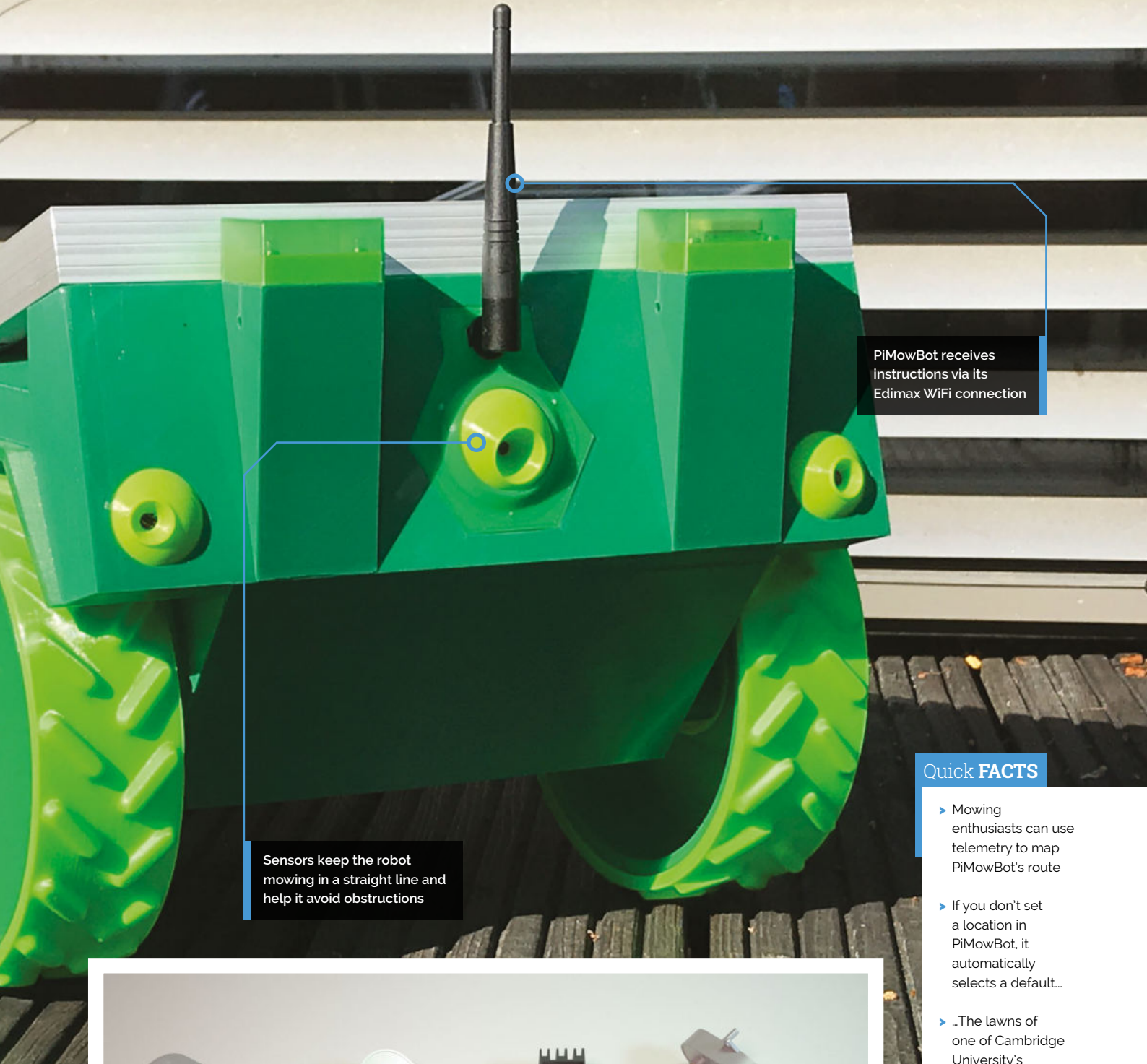
PiMowBot's solar panel takes advantage of the fact it's mainly used outdoors on sunny days

to help with the object modelling and 3D design (magpi.cc/pimowbotcase). One problem: Dirk is based in Schleswig-Holstein in northern Germany and Tim in Berlin, so the two collaborated via email. In fact, they've never met. Both have full time jobs, which meant keeping the project ticking along could be quite a challenge.

Dirk aimed to keep the build costs to around €300 – nearly two years later he estimates the project has cost him nearer €1500, though this includes the cost of two 3D printers and two previous prototypes. Now it's built, there are no significant running costs. "Solar power is free and the built-in hardware is robust and reliable and has no real wearing parts, except the battery," Dirk says.

False starts

The core parts are a Raspberry Pi Zero, Raspberry Pi Camera Module, and Witty Pi mini (a clock and



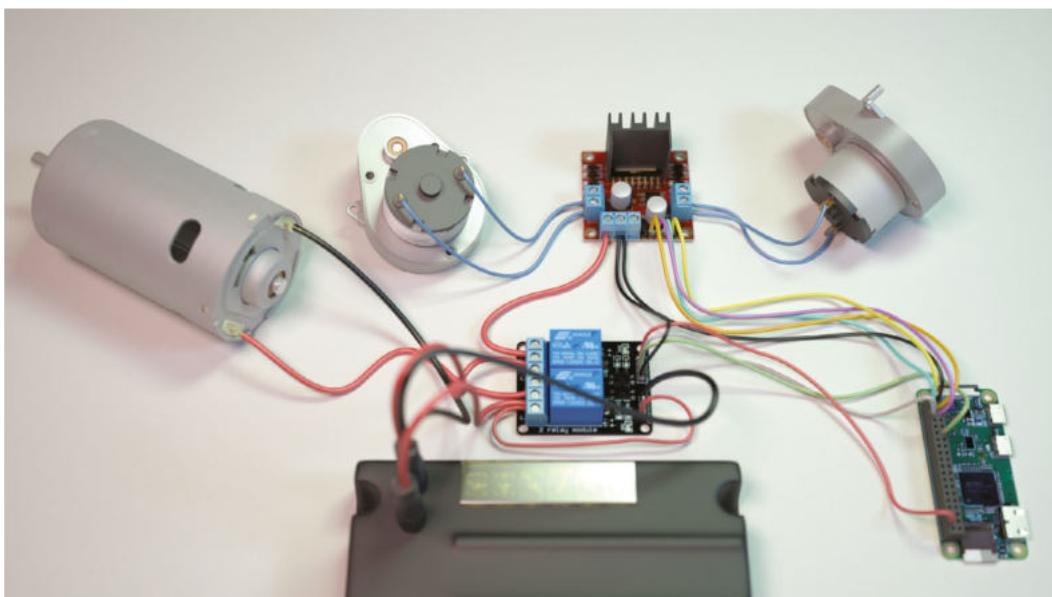
PiMowBot receives instructions via its Edimax WiFi connection

Sensors keep the robot moving in a straight line and help it avoid obstructions

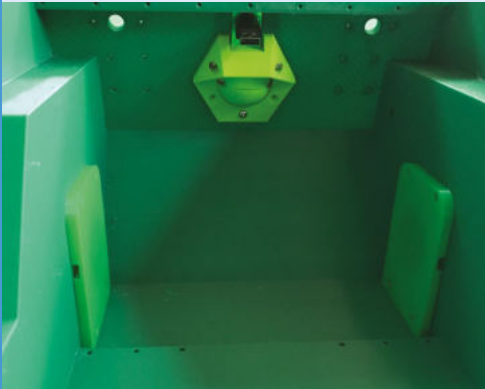
Quick FACTS

- ▶ Mowing enthusiasts can use telemetry to map PiMowBot's route
- ▶ If you don't set a location in PiMowBot, it automatically selects a default...
- ▶ ...The lawns of one of Cambridge University's colleges!
- ▶ PiMowBot is protected by a geofence theft deterrent
- ▶ It'll have instructions for the most efficient solar-charging alignment

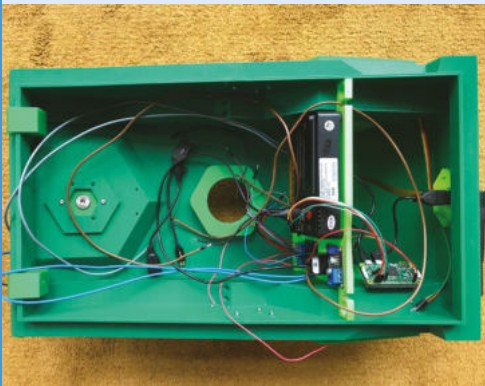
◀ PiMowBot's motor and L298N motor controller hooked up to Raspberry Pi Zero



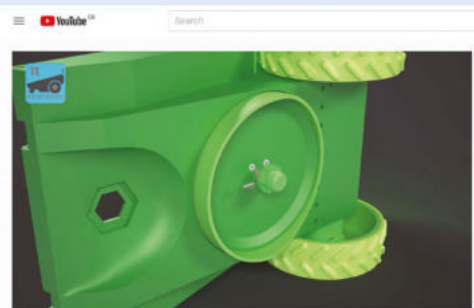
Make a mower



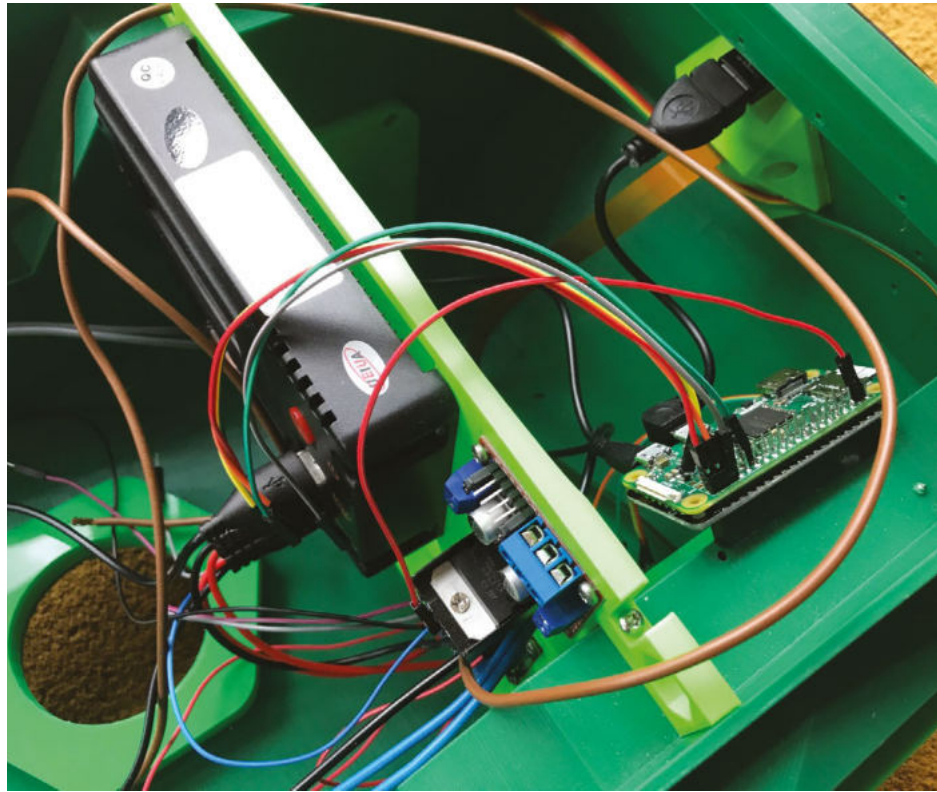
01 Download the 3D templates from magpi.cc/pimowbot or create your own. The PiMowBot software is supplied as a 1MB tarball. Setup instructions are at magpi.cc/pimowbotsetup.



02 Attach sensors, the L298N motor controller, and other hardware to Raspberry Pi Zero. The full parts list is at magpi.cc/pimowbotbom.



03 For help setting up the hardware for your Raspberry Pi Zero robot mower, see Tim's videos at magpi.cc/pimowbotyt.



▲ The 3D-printed housing has plenty of room for all the electronic parts

power management module) plus GPS, solar panel, battery, motor, and motor controller.

As well as the Raspberry Pi OS and the Witty Pi software, Dirk made use of standard lines of code from earlier versions of his robot mower. He coded most parts of PiMowBot in REBOL3 (magpi.cc/rebol) since he's been using the open-source functional programming language for nearly two decades. Python and GPIO Zero are used to communicate with the I²C devices, sensors, and motor control drivers.

“ Solar power is free and the built-in hardware is robust and reliable ”

Getting a precise orientation reading was a challenge due to the influence of magnetic north, but Dirk overcame this by getting PiMowBot to automatically go through a calibration process each time it's set up.

He wanted to keep things as straightforward as possible, experimenting with various motors and drivers such as a Waveshare Motor Driver HAT to control the gearing, braking, and acceleration.



- ◀ Sensors check the robot's precise location and confirm which way it's facing before mowing begins
- ▼ A Raspberry Pi Zero is the brain of the automated mower

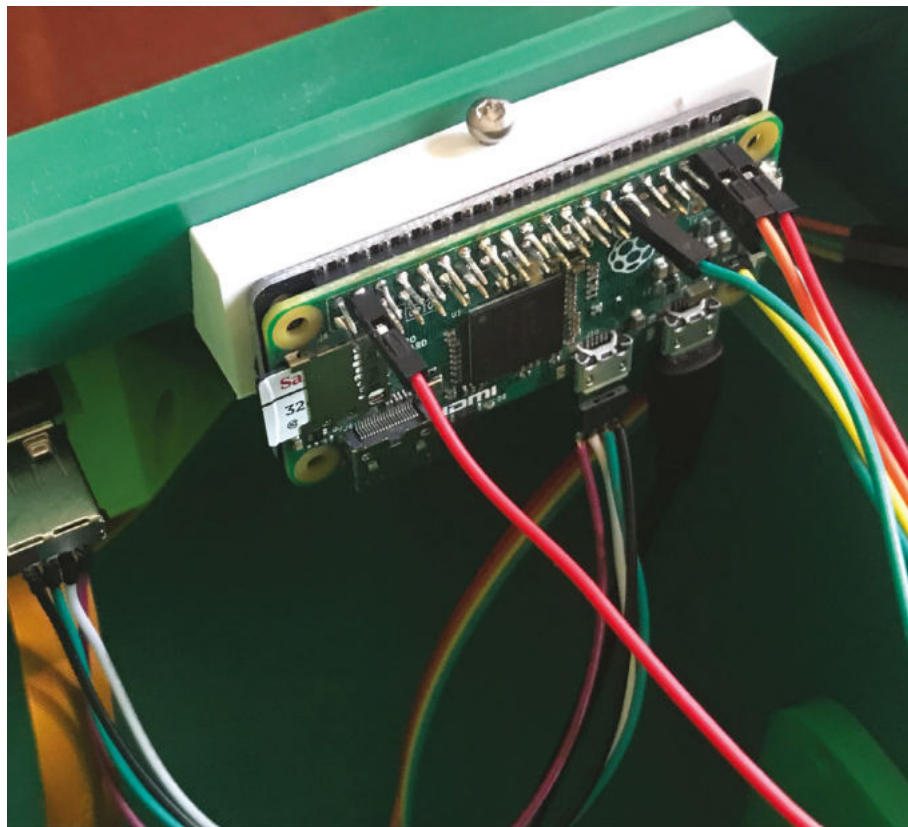
However, the geared motors drew too much current, prompting a switch to an L298N dual H-bridge module. Further refinements included adding a more powerful motor, which also meant Tim needed to redesign the 3D case.

The final version of PiMowBot was complete by the start of this year, with build instructions and the software all available from Dirk's website: magpi.cc/pimowbot.

Mow-tivational

"The PiMowBot has two operating modes: an autonomous operating mode without induction wires in the ground, and a remote control mode to directly control the lawn-mower robot via its web UI and the location and orientation sensors on the lawn," explains Dirk. He's used the Camera Module for previous Raspberry Pi projects and is keen to further refine the optical object/obstacle detection with real-world testing during the summer months.

Although PiMowBot was intended as a "sophisticated hobby project" to work on in his and Tim's spare time, Dirk is planning new models, including an XXL PiMowBot model and an intelligent system in which multiple PiMowBots operate alongside each other and share the mowing task. All hail our future mowing gods! [M](#)



OpenFlexure Microscope



© Brian Callaghan

MAKER

Dr Richard Bowman

Richard is a Royal Society University Research Fellow and Proleptic Lecturer at the University of Bath. He runs a research group primarily interested in microscopy automating lab experiments using open-source hardware and software.

openflexure.org

By focusing on the design of this open-source, 3D-printed microscope, Dr Richard Bowman's team is helping scientists to inexpensively detect disease, as **David Crookes** explains

Research-grade microscopes don't come cheap. They cost upwards of £30,000 for a decent model with a motorised stage – putting them outside the scope of many a pocket. As such, many attempts have been made to create low-cost alternatives, and one project – OpenFlexure – could revolutionise the detection of disease in developing countries.

Dr Richard Bowman is spearheading the creation of an open-source, 3D-printed microscope that's able to be adapted for use in labs, schools, and the home. He was inspired by his time as a Research Fellow in the NanoPhotonics Centre at the University of Cambridge, where an early attempt at such a project was falling short because most of the mechanism was based on linear bearings and metal rods.

"The bulk of the components weren't printable and I became curious as to how much of a microscope's mechanism you could print," he says. "My first version had basic focus control and an extension tube for a Raspberry Pi Camera Module to turn the stock webcam lens into a basic, but functional, microscope objective lens." The aim is to create an easily replicable scientific instrument.

Time to focus

Initially, Richard sought an ideal way of moving a sample around and picking a region to view. Rather than use sliding rails which require precise machining to be smooth, the eventual design was based on the flexibility of plastic: samples are placed on a table with bendable legs that allow for controlled focus and movement on the X and Y axes.

"It uses some fairly simple geometry to convert flexible hinges into linear motion," explains Dr

Joel Collins, who later joined the project at its home at the University of Bath. "It also means we can achieve really fine sample manipulation of tens of nanometres, for orders of magnitude cheaper than most commercial microscopes."

Unlike traditional microscopes, the project uses an upside-down design. The camera is at the base and the viewing lens is above with the light source at the top. It makes the microscope more stable. "You can arrange things so the sample is consistently close to being in focus when you place it on the microscope's stage, which is nice," Richard says.

Such work has helped to keep costs down. As a consequence, a student group that became WaterScope saw its potential for cheaply identifying early-stage bacterial contamination in water. Another student project demonstrated how

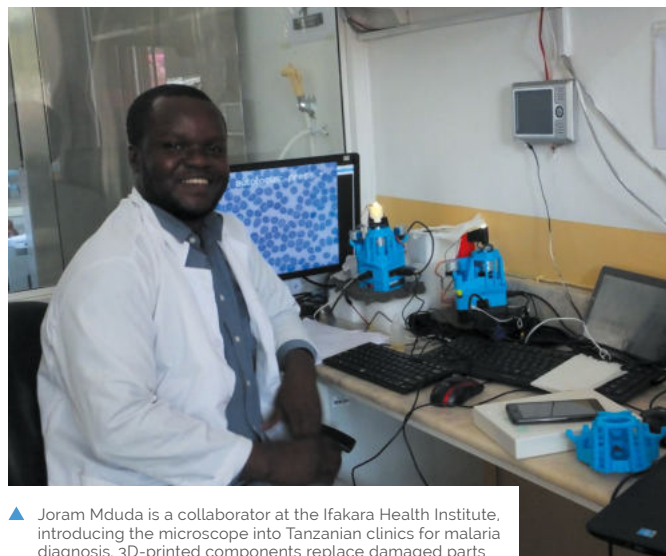


MAKER

Dr Joel Collins

Joel is a post-doctoral researcher at the University of Bath, primarily developing open-source software to control the OpenFlexure Microscope and automate collection of clinical data in Tanzania.

openflexure.org



▲ Joram Mduda is a collaborator at the Ifakara Health Institute, introducing the microscope into Tanzanian clinics for malaria diagnosis. 3D-printed components replace damaged parts



▲ Funding from the Royal Academy of Engineering and the Royal Society is being used to develop educational materials to enable the microscope to be used in schools

▼ Non-printed parts have been minimised by using a 'flexure' mechanism to move the stage, so the whole mechanism is printable



Quick FACTS

- ▶ The microscope is 'laboratory-grade'
- ▶ It's roughly the size of a bag of sugar
- ▶ Control it remotely over Ethernet or WiFi
- ▶ It's designed for easy replication
- ▶ More than 100 have been printed in Tanzania and Kenya

“We can achieve really fine sample manipulation of tens of nanometres”

it could be produced in Tanzania by STICLab. This led to a study which found the microscope could be used for malaria diagnosis.

Lighting the way

In creating the project, Richard's team has benefited from Raspberry Pi's bustling community. “There are great libraries for interacting with physical hardware, such as picamera and GPIO Zero,” says Joel.

“We created an HTTP API in Python and a graphical client in JavaScript that are both served by a Raspberry Pi,” he continues. “Since we have a full, powerful Linux machine in the microscope, we have many new ways to interact with the software.”

It's possible to create an offline, standalone microscope with a fully featured interface. “If you connect it directly to your laptop via Ethernet, you can control it 'headless' like that, or you could have a fleet of microscopes connected to a LAN, and our software will discover all the nearby microscopes and let you control them all from a single machine,” reveals Joel.

There no doubting its impact. OpenFlexure Microscopes have been recorded on every continent, including Antarctica. The biggest challenge, however, is documentation and communication. Richard says: “Building a piece of hardware is more involved than installing pre-built software and I have a new respect for anyone who works at IKEA – their instructions must have been so carefully tested and optimised.” **W**

The microscope uses three screws for fine control so that the sample can be moved along the X and Y axes and focused by moving on the Z axis

The microscope uses plastic flexures to ensure motion is free from friction and vibration; the condenser mount at the top houses a condenser lens and illumination LED

The eight-megapixel Raspberry Pi Camera Module v2 is used as the microscope's image sensor, with the electronics (including Raspberry Pi) housed in the base



Calendar-Controlled Busy Light

To prevent his daughter gatecrashing home-based work calls, Jim Bennett had a light-bulb moment, as **David Crookes** explains



Jim Bennett

Jim is an academic developer advocate at Microsoft, spending his days playing with IoT devices and writing code and documentation for students, faculty, and makers.

magpi.cc/ccbusylight

Working from home has many benefits: no commute, comfortable clothes, and copious amounts of coffee to hand. Throw a seven-year-old child into the mix, however, and knuckling down to some serious work can soon be hampered by an inquisitive mind. Jim Bennett encountered this problem when his daughter found herself off school for a long while. “I don’t mind her coming into my home office for a cuddle any time she wants to, but I don’t want to be recording a video or be on a call and have her running in to crack a poop joke,” he laughs.

As such, Jim looked for a way to let his daughter know that he’s hard at work. “An on-air light seemed the obvious solution,” he says. “I’ve read of others having such a light set up outside their office and, because I’d been playing a lot with NeoPixels and other lights, it was foremost in my mind.”

Setting the mood

The project fell together from that point on. “A friend had given me a Pimoroni Mood Light which I hadn’t used, so that provided me with an easy choice,” he recalls. “It’s a great piece of kit that

includes a Raspberry Pi Zero W, a Unicorn pHAT with programmable RGB LEDs, and an acrylic stand that diffuses the LEDs into a lamp shape.”

It involved some soldering of Raspberry Pi Zero W headers and the Unicorn pHAT—“I’m really bad at that,” he confesses—but from that point on, the bulk of the project involved getting the assembled device to talk to his calendar. That way, he wouldn’t need to keep getting up to manually turn the light on: the idea was that it would automatically flick on to red during a scheduled meeting and show green when he was free.

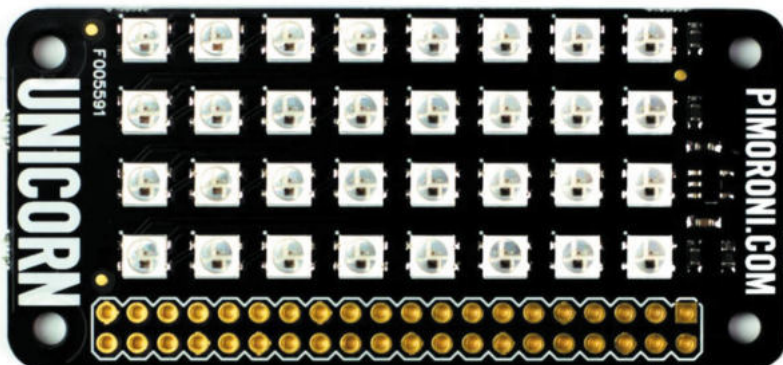
“I don’t want to be recording a video and have my daughter running in to crack a poop joke”

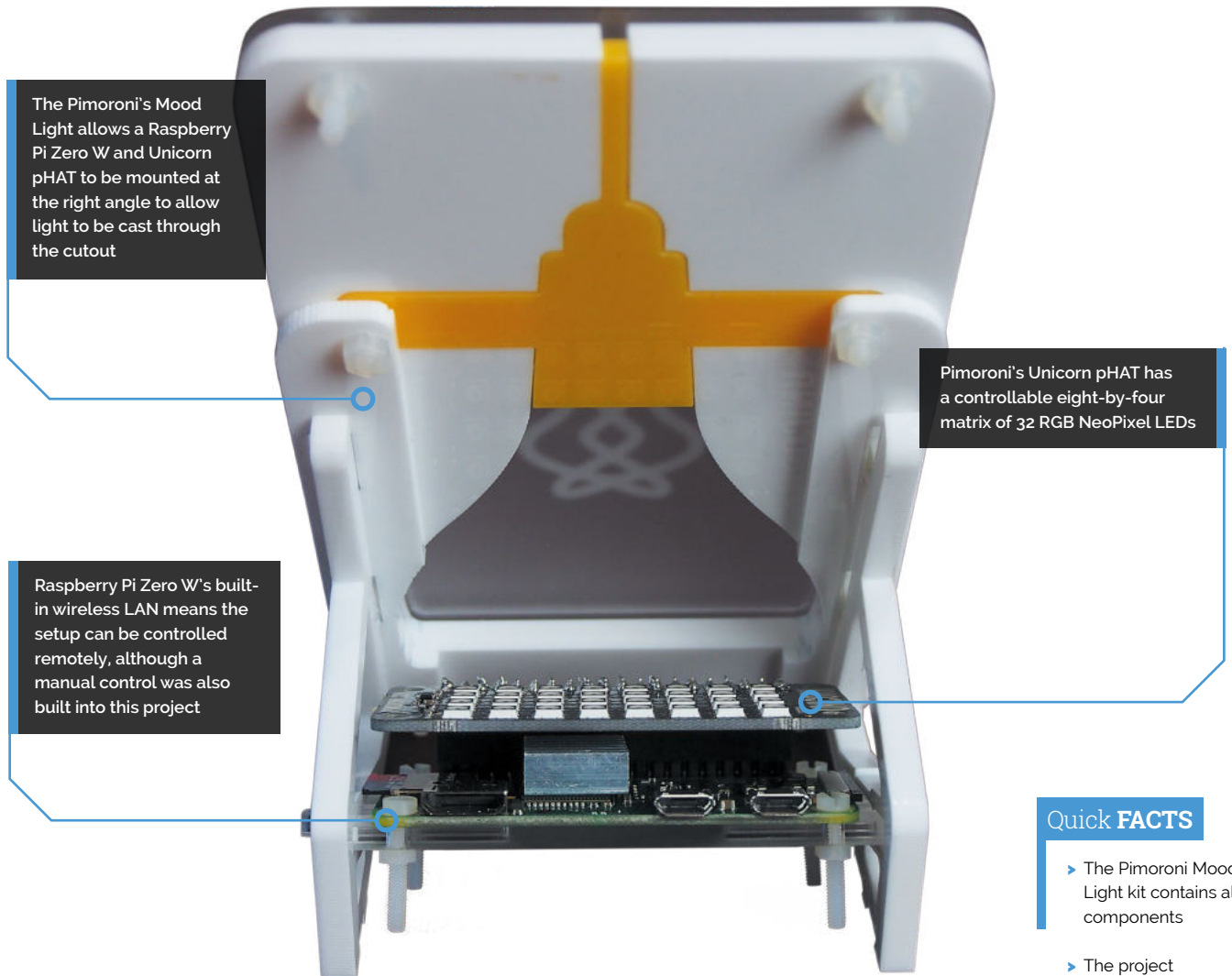
“I do a lot of work with Azure IoT Central, an IoT software-as-a-service platform from Microsoft,” Jim reveals. “It allows you to define what you want a device to do without any code and it let me create a command that would send a colour value to Raspberry Pi Zero W.”

Calendar control

Jim also wrote 100 lines or so of Python code so that Raspberry Pi Zero W would connect to Azure IoT Central, listen for the command, and change the colour of the Unicorn pHAT.

Azure Logic Apps was used to control the device from his Office365 calendar. “This is another service from Microsoft that allows you to build logic with no code, using drag-and-drop programming,” he says. A feature called Connectors polls his calendar for upcoming





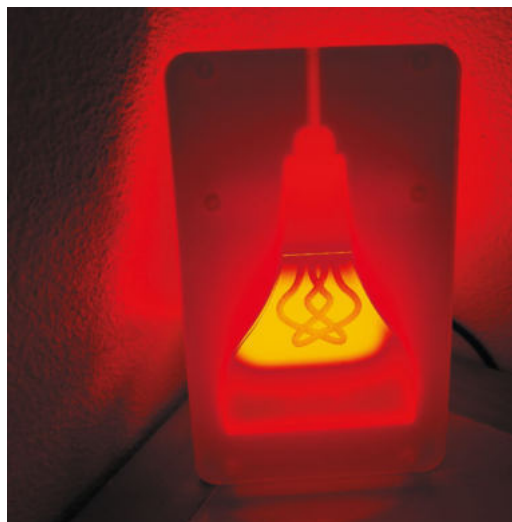
Quick FACTS

- ▶ The Pimoroni Mood Light kit contains all components
- ▶ The project connects to an Office365 calendar
- ▶ Internet connectivity is therefore crucial
- ▶ Some soldering will be required
- ▶ You'll need a Microsoft Azure account

events, connects to IoT Central, and turns the light to red when an event starts, then green when it ends.

Yet he still wanted manual control in case he had an impromptu call or a meeting ended early. "For this I used another logic app to make a web endpoint I could call from anywhere that also connected to IoT Central," he explains. "Then to call this, I used a Flic button which could be operated via my phone to call the web endpoint."

So far, the setup has worked a treat and being able to get it up-and-running quickly has saved many potentially embarrassing moments. "My daughter has come in for a cuddle during meetings but she does it knowing that I'm in a call so she's quiet," he says. "But there have been no poop jokes when I'm recording or on a call, so it seems to be working." **M**



- ◀ A red glow means it's no go for Jim's daughter – although the temptation to pop into his home office for a cuddle remains strong

Mind Patterning

Michael Darby doesn't just admire superheroes and sci-fi stars – he's doing his best to become one. **Rosie Hattersley** asks why



Mike Darby

MAKER

IT QA engineer and a lifelong video games and electronics fan, Mike would love to work with Elon Musk and try and build a time machine.

magpi.cc/314reactor

Michael Darby, aka **314reactor**, has been having ideas for electronics projects for as long as he can remember. “As a kid I’d try to make things (with poor results) by badly soldering up circuits. I was just too young to really make it work,” he says.

A lifelong video games and technology fan, Michael picked up his first Raspberry Pi in 2013 and was excited to play around with it and see what he could do with it. After a few false starts, he realised his inventiveness could finally be given free rein. His first big project was a Windows 98 Wrist Watch. Hop forward in time three years and he’s lost count of the number of Raspberry Pi boards he possesses.

Since then he’s made several Raspberry Pi projects, including Spider-Man specs – hence his profile pic. His Mind Patterning headset is intended “to kick off a much larger project using mind-reading capabilities.”

“With more tweaking and more data, it could be used to interact with the hardware”



▲ Training the mind-patterning headband to discern whether Mike’s happy or unhappy (will it still work when he finally becomes a robot, though?)

The project reads brainwave data with a MUSE headband (magpi.cc/brainheadband) and uses TensorFlow machine learning to train a model on the data to recognise a relaxed or non-relaxed mind state.

Mind over matter

Michael used a Raspberry Pi 4 for the Mind Patterning project “because it allows lots of applications to run simultaneously.” The Bluetooth setup required for Raspberry Pi to import data from the headset turned out to be one of the trickier aspects of the build since the latter is really designed to work with iOS or Android.

“Using Raspberry Pi was essential for the TensorFlow tech required for training a neural network on the recorded EEG data,” he explains. “The code is where the design and development was [see his GitHub repo:

magpi.cc/mindpatterning]. I used some libraries from elsewhere in order to get the EEG data logged and accessed. So many people use Raspberry Pi and it ensures that there is always support and information out there to learn from and assist,” he notes.

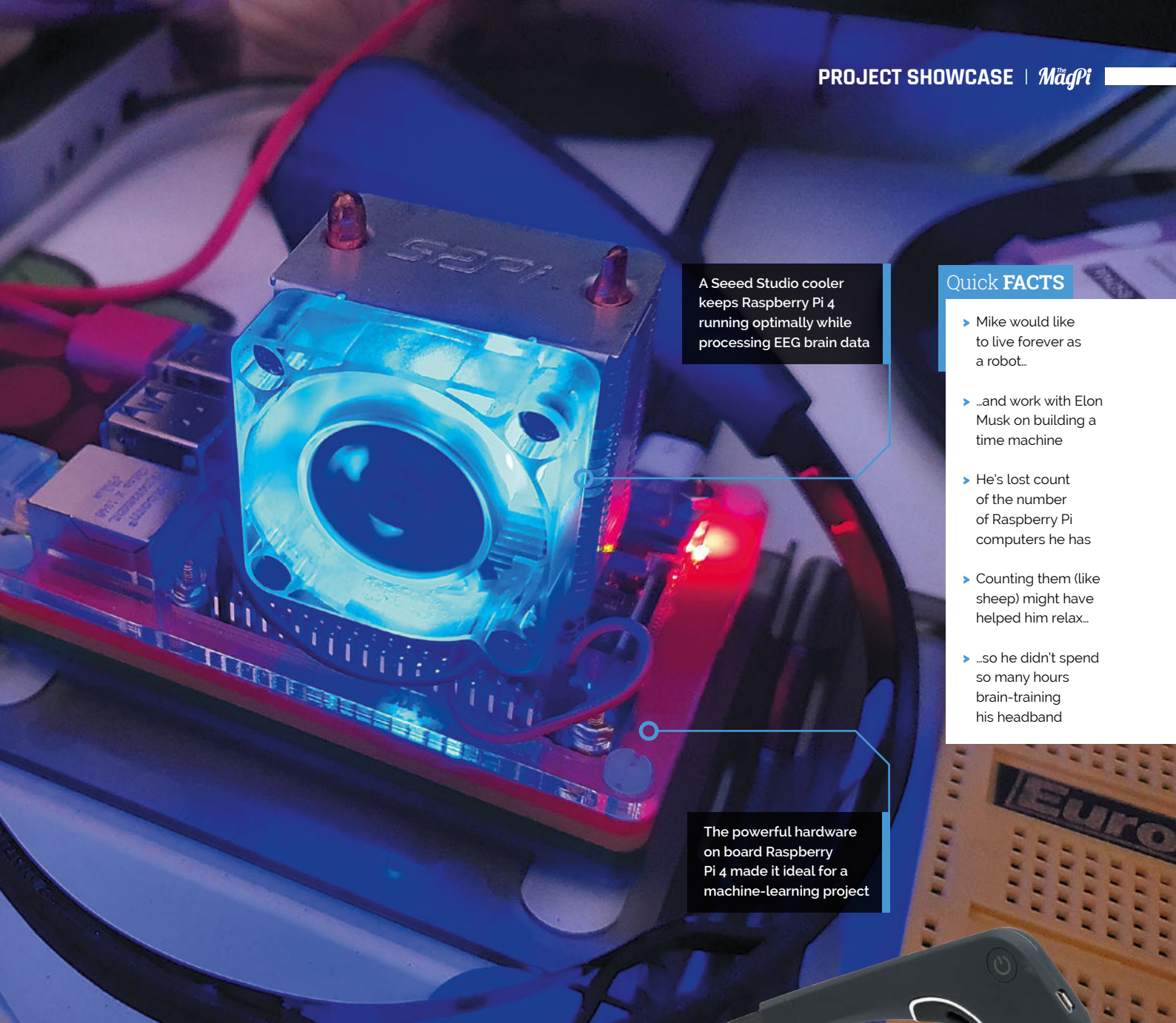
Raspberry Pi 4’s flexibility and processing power meant Michael needed little else to realise his project. “It has so much hardware on board, including the power for the machine learning, which is where the multicore and lots of RAM really helped,” he says.

Trial and error

“Initial testing of the model training and inference was done with the PipelineRunnerTest.py file,” explains Michael. “This creates artificial sets of training and neural network testing data. It calls



The Muse EEG Bluetooth headset captures the wearer’s brainwaves



A Satechi cooler keeps Raspberry Pi 4 running optimally while processing EEG brain data

Quick FACTS


- ▶ Mike would like to live forever as a robot...
- ▶ ...and work with Elon Musk on building a time machine
- ▶ He's lost count of the number of Raspberry Pi computers he has
- ▶ Counting them (like sheep) might have helped him relax...
- ▶ ...so he didn't spend so many hours brain-training his headband

The powerful hardware on board Raspberry Pi 4 made it ideal for a machine-learning project

CSV preprocessing (which loads and trains the model), then runs that against the test data.”

This approach allowed him to see quickly how well the model would work if given real-world data, and to make adjustments. “I was trying to get the raw EEG sensor data at first, but then realised I needed to get the data processed into alpha/beta/theta wave data before passing to the neural net.”

Michael hopes to develop the project to read more mind states, possibly even emotions, and sees future potential for the mind-patterning idea in simulating human responses via a neural network.

“With more tweaking and more data, it could be used to interact with the hardware without having to manually press buttons, but I’m sure Elon Musk’s Neuralink is going to be far, far more effective than anything I could ever do,” he says. 



- ▶ The Muse headset is really intended for use with iPhones or Android devices, so Mike had to write new Raspberry Pi OS code

- ▶ One of Mike's other projects recreates the EDITH glasses from *Spider-Man: Far From Home*



LEDWeatherBoard



Diego Braga

A solution architect from fair Verona, Italy. He's been featured in *The MagPi* before, along with his pet goldfish.

@braghetos

Check up on the local weather forecast with this giant LED display. **Rob Zwetsloot** grabs his umbrella

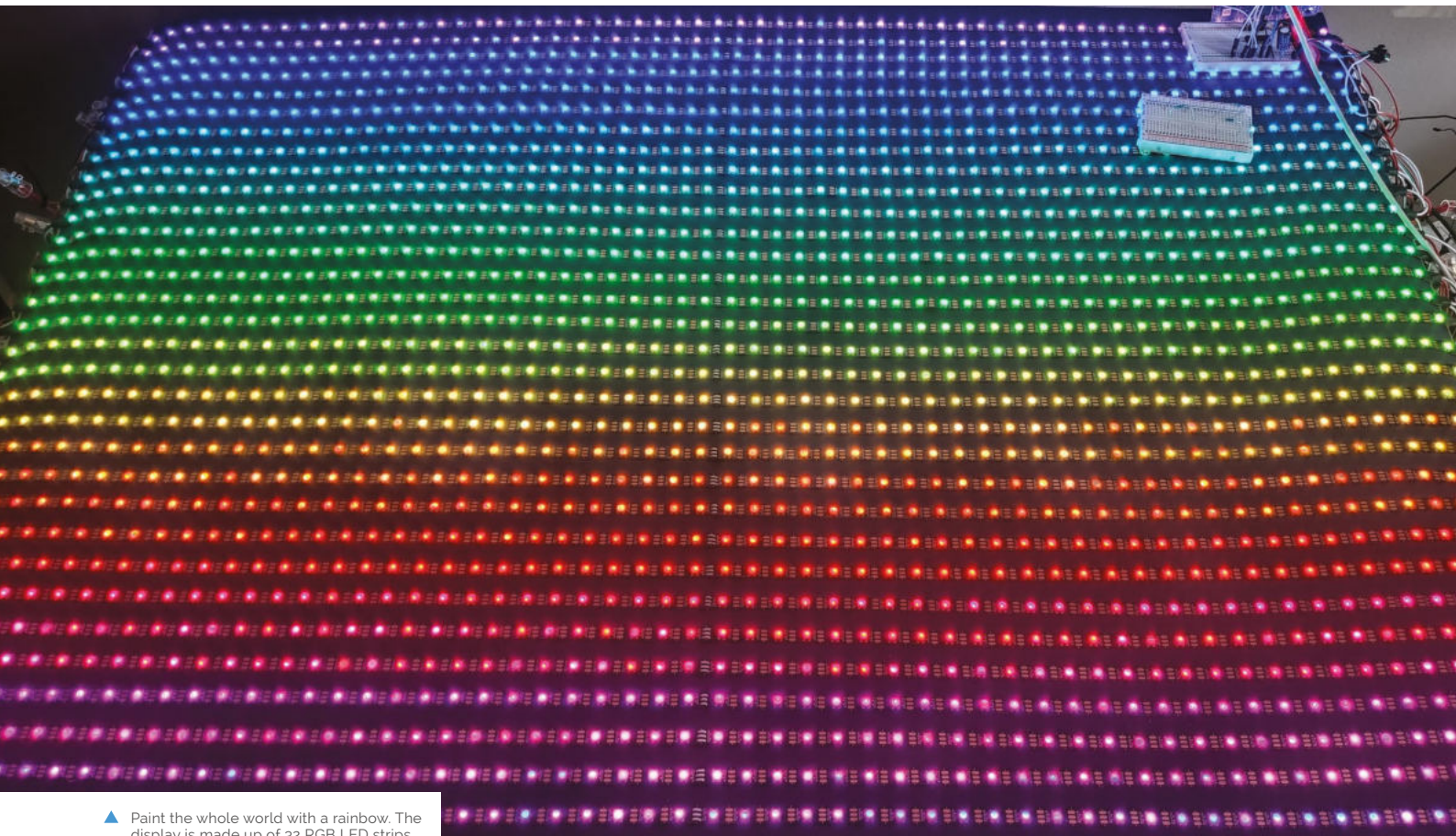
The weather. It's a stereotypical, and very real part, of British conversation, and a classic example of using online information in your programming. Keeping the classic idea in mind, Diego Braga paired the concept with a giant board full of LEDs.

"I had the idea talking with some friends that are always checking the weather forecast," Diego tells us. "My idea was to create something that could be even [as] good-looking as a piece of furniture, something that reminds of an 8-bit era, but with a simple layout that could fit any wall in any situation."

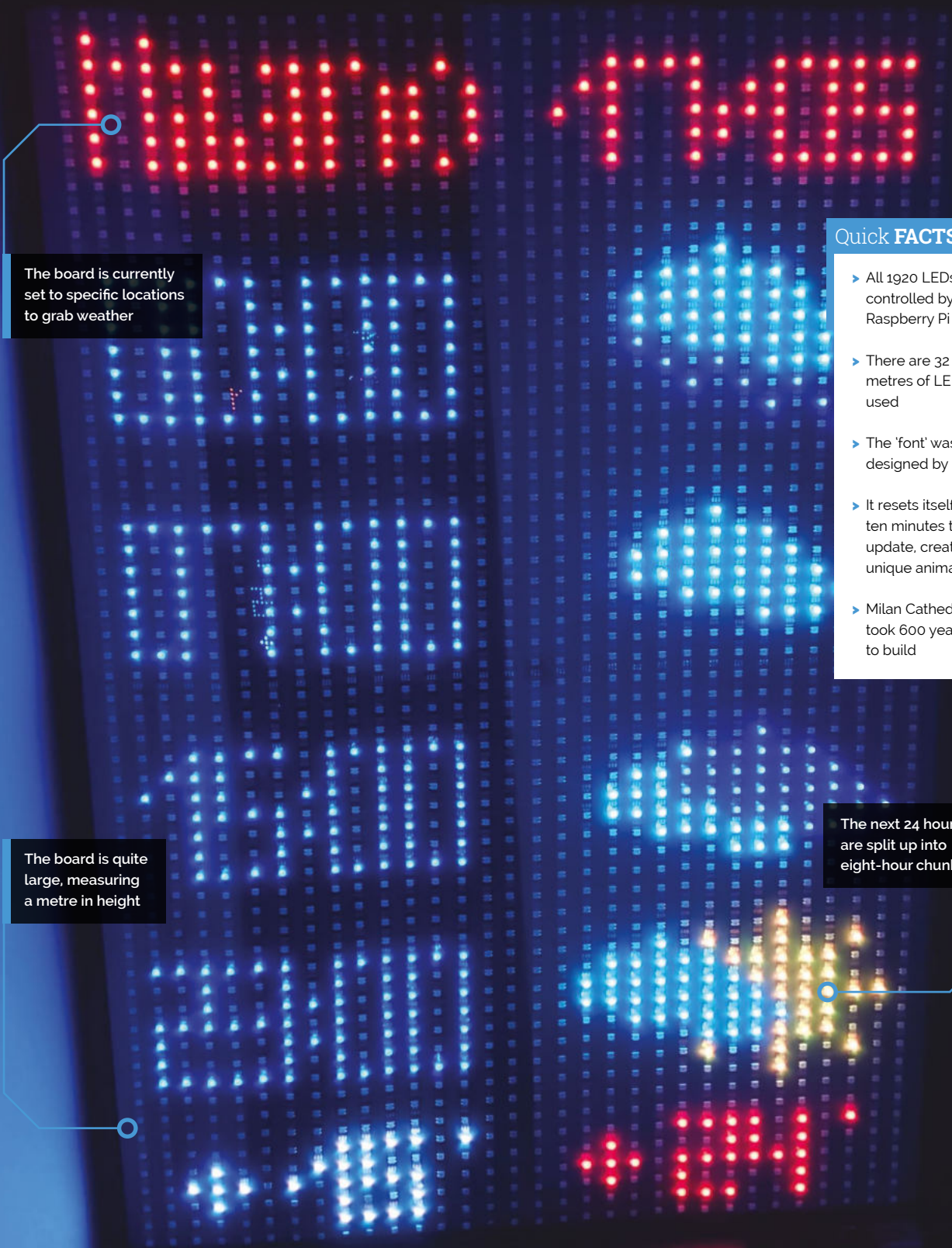
What Diego ended up with was a 70×100 cm board, with 1920 RGB LEDs split up over 32 LED strips. Each individual LED is controllable, including what colour it displays as.

Today's forecast

"Raspberry Pi is the perfect device to both call a public API for weather forecast and drive via GPIO the 32 metres of LED strips," Diego explains. "I can access my Raspberry Pi via SSH and control the logging of the workflow that I implemented using Node-RED and Docker, in order to have a repeatable and consistent



▲ Paint the whole world with a rainbow. The display is made up of 32 RGB LED strips



The board is currently set to specific locations to grab weather

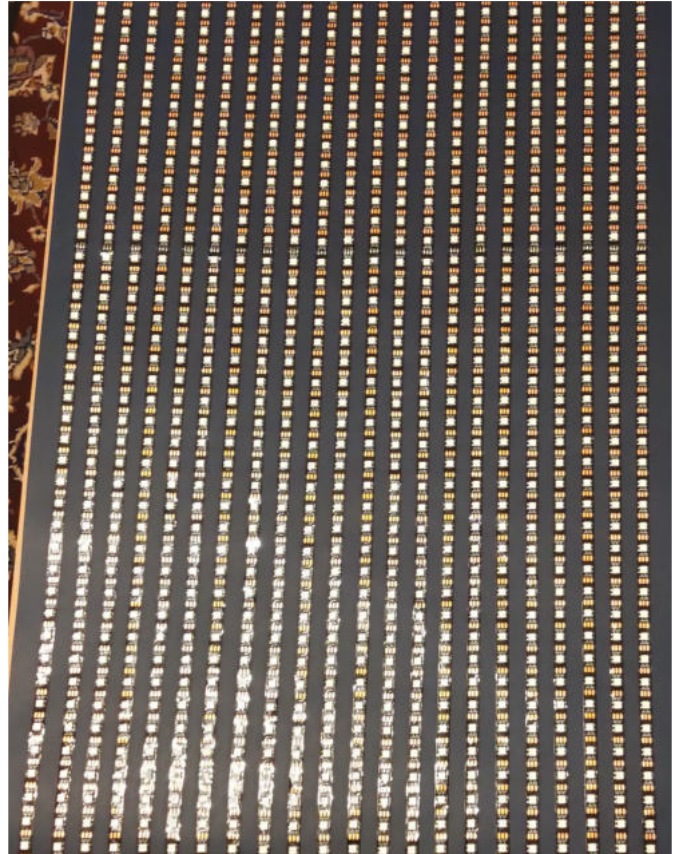
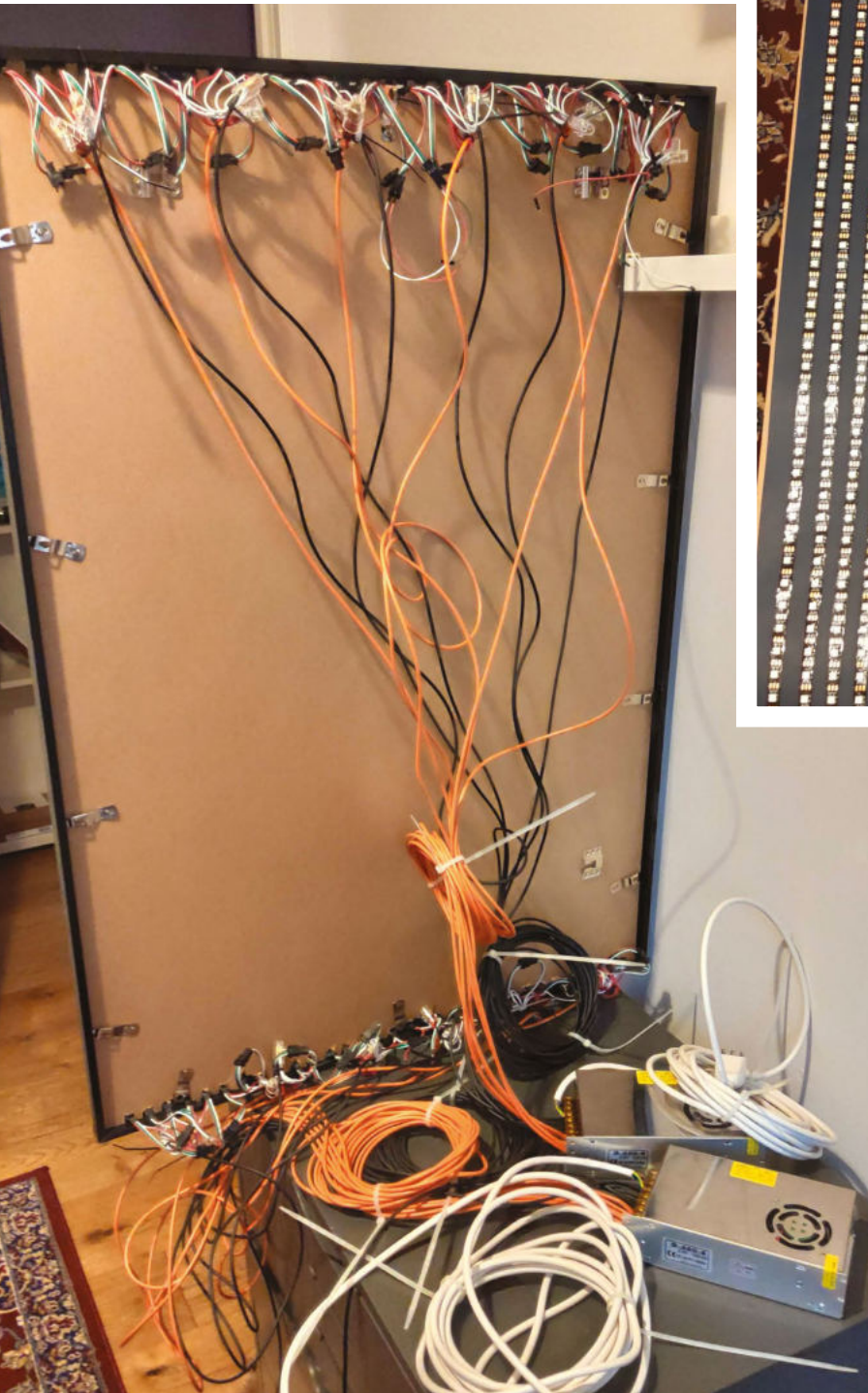
Quick FACTS

- ▶ All 1920 LEDs are controlled by one Raspberry Pi
- ▶ There are 32 metres of LEDs used
- ▶ The 'font' was designed by Diego
- ▶ It resets itself every ten minutes to update, creating a unique animation
- ▶ Milan Cathedral took 600 years to build

The board is quite large, measuring a metre in height

The next 24 hours are split up into eight-hour chunks

▼ It's not pretty behind the board

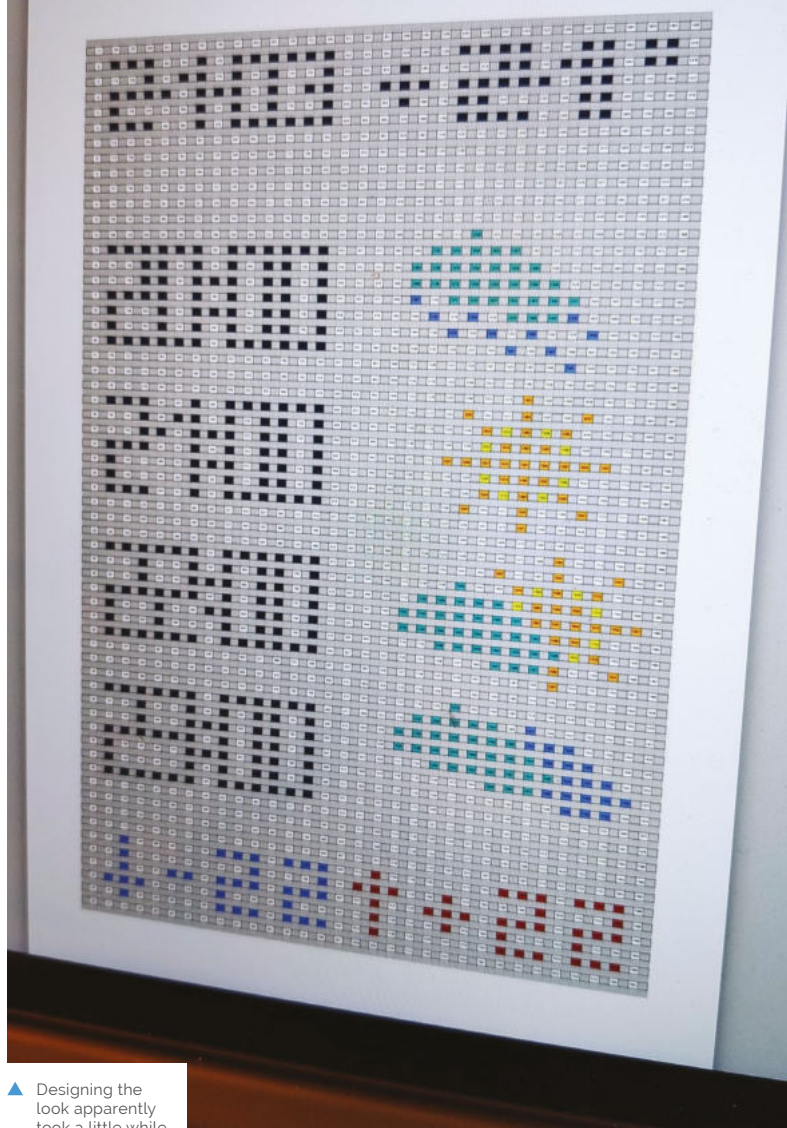


▲ There's a gap of just over a centimetre between strips

environment to provide to anyone interested in using my Docker image.”

While it may look and sound a little simple, Diego did have a few issues with ensuring enough current can be supplied to the LEDs. “If all the 1920 LEDs are set to white, then I need to provide two power supplies, each providing 5 V 50 A 200 W,” he says. “I chose the WS2812B-eco LEDs strips that require less current but are more sensitive to bad connections, so I had to use a resistor between the initial input of the 32 LED strips and the pin of the Raspberry Pi.” He even had to use a level-shifter to boost the signal from the GPIO pins to the input of the LED strips.

“I had to try different libraries to drive the LEDs because initially I experienced the flickering of the LEDs,” he adds. “Raspberry Pi has the capability to drive two PWM pins, so I had to split the 32 metres in two segments. After all this hardware setup, now the LEDWeatherBoard is a working project with the level of perfection that I require from my projects!”



▲ Designing the look apparently took a little while

“LEDWeatherBoard is a working project with the level of perfection that I require from my projects”

Predicting tomorrow

Diego isn't quite finished with the project yet, wanting to add positioning hardware and interactivity.

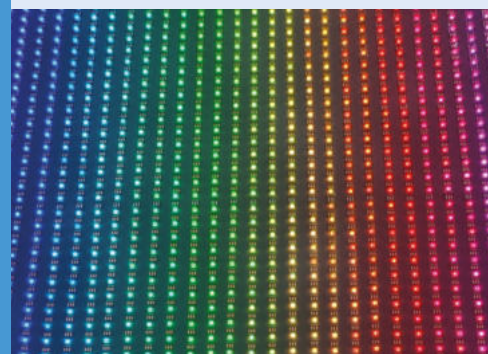
“I also programmatically change the LED brightness if the request to the weather API is made after sunset or after sunrise,” he reveals. “I would like to change this last hard-coded logic using a photoresistor that can read the environmental brightness and adjust [it] accordingly.”

We think it's a great project, and it turns out his friends agreed – one of whom had it installed in his hotel lobby in the centre of Milan, near the Duomo (cathedral). Make sure it's not raining before you head out. [M](#)

Displaying the weather



01 “A weather API is called every ten minutes and acquires the next 24-hour forecast for a city. The graphic I choose for the LED board shows the city that is forecasted, the date, the min and max temperature, and four time rows that are dynamically calculated by dividing the 24-hour forecast (into four parts). So, if I'm calling the weather API at 8:00am, I'll show four rows with 8:00, 16:00, 22:00, 8:00.”



02 “An icon is calculated based on the weather forecast. The right position to show the icons is calculated dynamically, so if I want to shift an icon or change how icons are composed, I can change the offsets of my initial positions and everything works great.”

03 “A 1920-element array is calculated, so each LED is lit reading the content of the relative array element. Once the LED wall is ready to be updated, I choose an animation that refreshes the hours and icons [for the] weather forecast from the left to the right of the wall.”

SUBSCRIBE TODAY FROM ONLY £5

SAVE UP TO 35%



Subscriber Benefits

- ▶ **FREE Delivery**
Get it fast and for FREE
- ▶ **Exclusive Offers**
Great gifts, offers, and discounts
- ▶ **Great Savings**
Save up to 35% compared to stores

Rolling Monthly Subscription

- ▶ Low monthly cost (from £5)
- ▶ Cancel at any time
- ▶ Free delivery to your door
- ▶ Available worldwide

Subscribe for 12 Months

£55 (UK)	£90 (USA)
£80 (EU)	£90 (Rest of World)

Free Raspberry Pi Zero W Kit with 12 Month upfront subscription only (no Raspberry Pi Zero W Kit with Rolling Monthly Subscription)

☎ Subscribe by phone: **01293 312193**

📍 Subscribe online: **magpi.cc/subscribe**

✉ Email: **magpi@subscriptionhelpline.co.uk**

JOIN FOR 12 MONTHS AND GET A

FREE Raspberry Pi Zero W Starter Kit

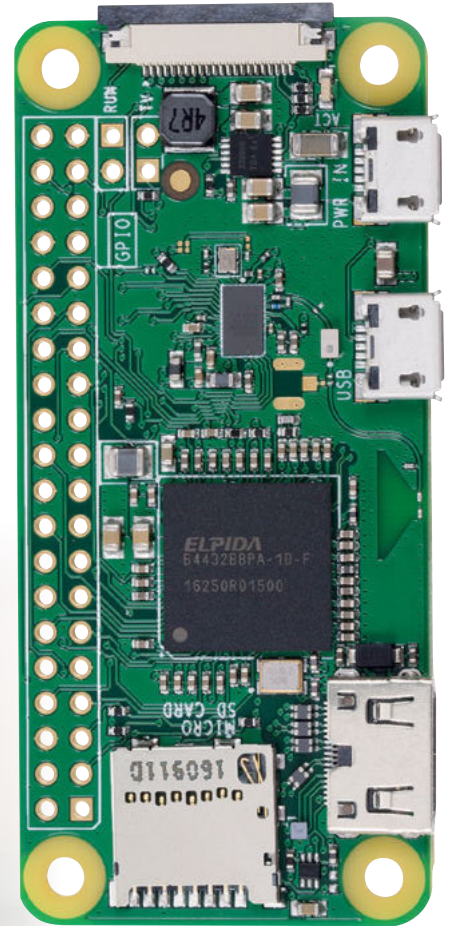
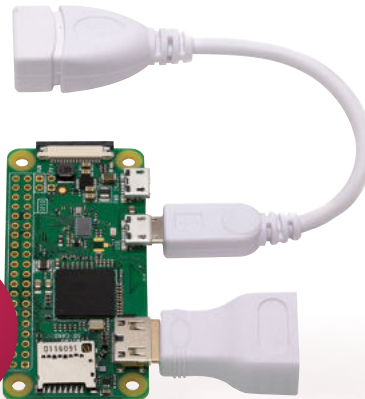
WITH YOUR SUBSCRIPTION

Subscribe in print
for 12 months today
and you'll receive:

- ▶ Raspberry Pi Zero W
- ▶ Raspberry Pi Zero W case with three covers
- ▶ USB and HDMI converter cables
- ▶ Camera Module connector

Offer subject to change or withdrawal at any time

WORTH
£20



Buy now: magpi.cc/subscribe

SUBSCRIBE on app stores

From **£2.29**

Available on the
App Store

GET IT ON
Google Play

25

NO-PROBLEM PROJECTS

Eager to put your Raspberry Pi to good use, but don't know where to begin? Here are some inspiring projects for you to build right away. By **David Crookes**

Raspberry Pi has bags of potential – a quick flick through the pages of *The MagPi* magazine provides evidence of that. But while many makers get stuck into projects, other people buy our favourite single-board computer and struggle to build anything with it. If you've got a Raspberry Pi, then we're here to help. We've collated a bunch of quick but inspiring projects that are ideal for absolute newcomers, easing you over the hurdle of having nothing to make. You'll soon be embarking on your very own maker journey.

01 Build a desktop PC

Raspberry Pi 4 heralded a coming-of-age for our favourite single-board computer. Grab the latest Raspberry Pi 4 model and you'll have a device you could easily use each day as a powerful desktop PC. Connect a keyboard, mouse and monitor, install Raspberry Pi OS to a microSD, card and plug in a power supply. Then discover amazing applications such as the LibreOffice suite, Minecraft Pi,

and the open-source Chrome-based browser, Chromium. Raspberry Pi 4 has a boosted GPU, so it's a powerhouse in the graphics department too, making it ideal for photos and video. You can even give your eyes a treat and connect two monitors.

Take a look at *The MagPi* #85 for an in-depth feature on using Raspberry Pi 4 as a desktop PC. magpi.cc/85



Tip!

Thanks to the optimised nature of Raspberry Pi OS, the entry-level Raspberry Pi 4, with 2GB RAM, makes a great desktop PC.



02 MAKE A GAMES CONSOLE

Install software such as RetroPie or Lakka and you can emulate a host of classic consoles. Play retro games stored as ROM files by connecting a USB controller and consider taking your nostalgic setup to the next level by popping Raspberry Pi into a case that resembles a past machine.

magpi.cc/retropie

Tip! Get legal ROMs

It's illegal to download copyrighted ROMs from the internet, but check out our list of legit games you can play.

magpi.cc/legalroms

03 SET UP A MEDIA PLAYER

Since Raspberry Pi can play video and music, you're able to put it to use as a versatile media player. VLC is the default, open-source platform offering accelerated video playback, but you can also use LibreELEC which has been designed to run the home media centre software, Kodi. Use Raspberry Pi Imager (magpi.cc/imager) to install LibreELEC to a microSD card. And read *The MagPi* magazine issue 87 for everything you need to know about setting up a media player.

magpi.cc/87



Three Starter Kits



04 OFFICIAL RASPBERRY PI 4 DESKTOP KIT

Everything you need to get started is in the official Desktop Kit: Raspberry Pi 4 Model B, the official keyboard and mouse, a couple of HDMI cables, a power supply, case, and microSD card. Even the official *Beginner's Guide* book.

magpi.cc/desktopkit

05 OKDO RASPBERRY PI 4 4GB MODEL B STARTER KIT

This kit is the perfect introduction for newbies. It includes a Raspberry Pi 4, cooling case with separate fan, and a switched power cable.

magpi.cc/okdostarter



06 RASPBERRY PI 4B STARTER KIT

This PiShop.us starter kit is packed with essentials. It comes with Raspberry Pi 4, HighPi case and heatsink, USB-C power supply, 16GB microSD card, and the cables you need.

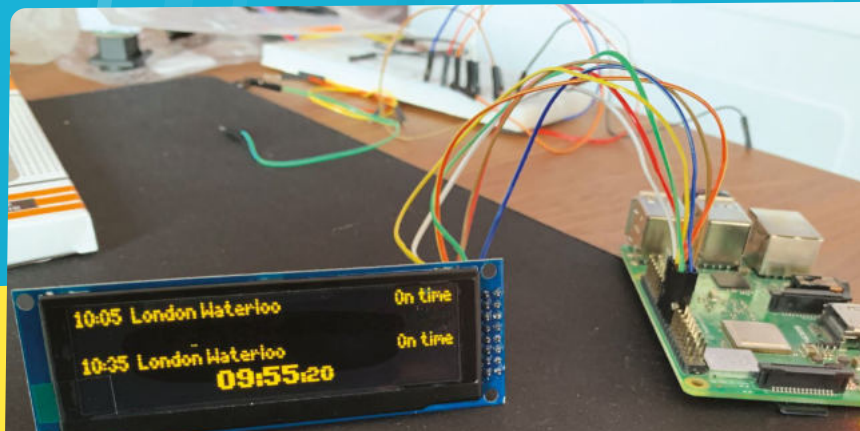
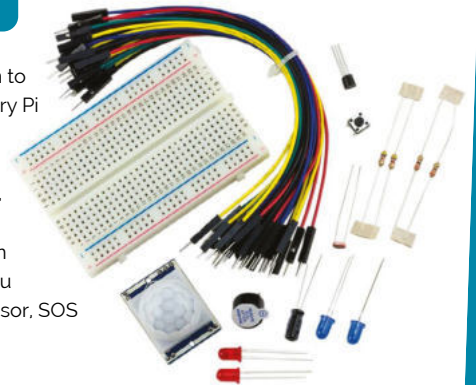
magpi.cc/pishopstarter

Easy electronics

07 BEGIN WITH THE BASICS

If you want a gentle introduction to electronics, ModMyPi's Raspberry Pi YouTube Workshop Kit contains a transparent breadboard, LEDs resistors, buzzers, buttons, sensors, and jumper wires for a series of projects explored in ten YouTube videos. They'll show you how to make a temperature sensor, SOS buzzer, and much more.

magpi.cc/pihutyoutube



08 DISPLAY LIVE TRAIN TIMES

Want to be kept informed of any train delays or cancellations before you leave the house? Chris Hutchinson's train departure board uses TransportAPI, some Python code, and an inexpensive 256x64 SSD1322 OLED screen to display real-time departure information. You can even download and use some authentic-looking fonts for that true platform feel.

magpi.cc/trainscreen



09 A NEW WHOOPEE CUSHION

Parp noises never fail to make us giggle, but the sound from a standard whoopee cushion isn't quite satisfying enough. Grab some paper plates, foil, crocodile clip cables, male-to-female jumper leads, and a pair of speakers, as well as other assorted bits and bobs, and let rip with any sounds you wish to inflict.

magpi.cc/whoopi

10 REFLECT WITH A MAGIC MIRROR

A beginner's project? Actually, the biggest hurdle in creating an internet-connected mirror is time, not technique. We made one in issue 90 (magpi.cc/90), and the electronics comprise a monitor, Raspberry Pi, and some cables. Download the code, and make something magical.

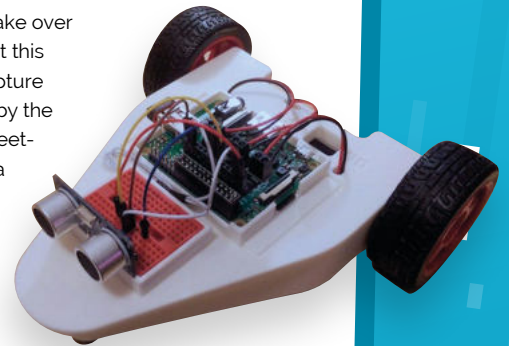
magpi.cc/buildmagicmirror



11 MAKE YOUR OWN ROBOT

Robots aren't quite going to take over the world... at least not yet. But this robot-making kit is sure to capture your imagination. Developed by the Cambridge Raspberry Jam meet-up, it requires you to provide a chassis (or you could use the box!), but otherwise most of what you need to bring your creation to life is included.

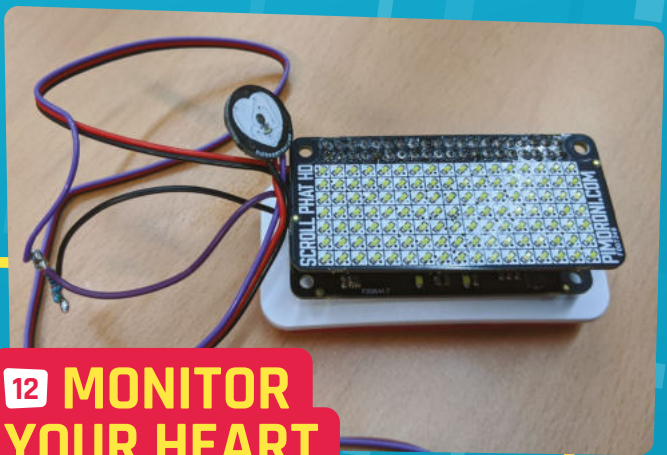
magpi.cc/edukit3



12 MONITOR YOUR HEART

Keep a check on your ticker with this simple heart rate monitor using the Enviro pHAT from Pimoroni, as well as a Pulse Sensor Amped. Put together by pupils at Queen Margaret's School in York under the steer of teacher Jon Witts, it measures live heartbeats and displays the results on a Scroll pHAT HD.

magpi.cc/heartate



Gaming kits



13 RETRO GAMING KIT

This easily assembled bundle includes two SNES-like USB controllers, a Raspberry Pi 4, and an aluminium case designed to disperse heat for when those gaming sessions become a little intense.

magpi.cc/retrogamingkit



14 PIMORONI PICADE

Original arcade cabinets are big old beasts, but the Picade mini-cabinet kit brings the spirit of the coin-ops into your home. Use the robust joystick and buttons to blast through some retro games on the hi-res 4:3 display.

magpi.cc/picade

15 RETROFLAG GPI

Sporting a classic look, this kit includes a clear 2.8-inch screen and a speaker controlled by a volume wheel. All you need is a Raspberry Pi Zero, a microSD card, and some batteries.

magpi.cc/gpicase



Smart home projects

16 Develop an intelligent speaker

Add powerful voice control to Raspberry Pi with Google's DIY intelligent speaker. The AIY Voice Kit allows you to converse with Google Assistant and even convert spoken commands into text via the Google Cloud Speech-to-Text service. All the components come in the box and it only takes a couple of hours to assemble.

magpi.cc/aiyvoice



17 STREAM AND RECORD LIVE TV

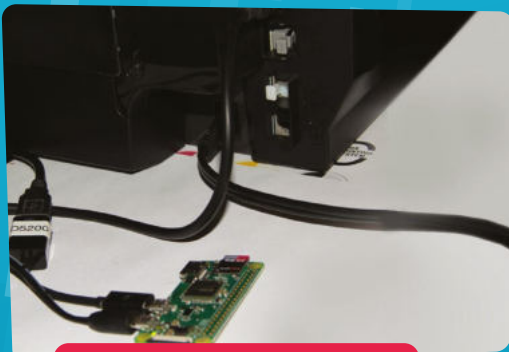
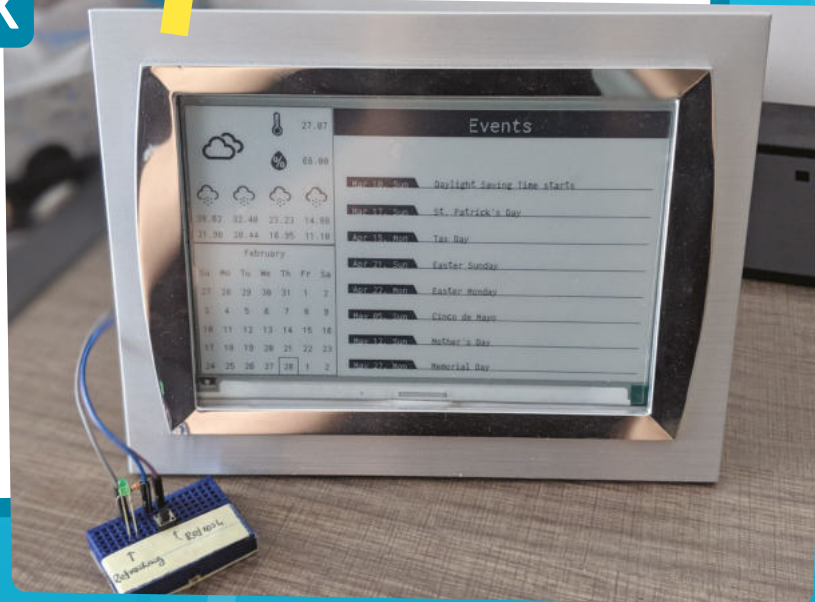
Connect an official TV HAT to Raspberry Pi and you'll be able to decode live TV signals and stream them to remote devices thanks to the built-in Sony CXD2880 tuner. More than that, you can connect a USB stick or a hard drive and create a digital video recorder so that you don't miss your favourite shows.

magpi.cc/tvhat

18 PRODUCE A SMART E-INK CALENDAR

E-ink displays draw less power than LCD screens. Perfect then for presenting this always-on calendar that updates once an hour, or at the press of a button. Zonglin Li has infused the front-end with numerous widgets and an LED lights up when the screen is being refreshed.

magpi.cc/einkcalendar



19 CONVERT A PRINTER TO WIRELESS

If you have a USB printer you'd like to network wirelessly, you can cut the cables courtesy of a wireless-LAN enabled Raspberry Pi computer – a Zero will do the job well. The project allows you to print from any device.

magpi.cc/wirelessprinter

20 CREATE A CLAP-POWERED SWITCH

Master this simple skill and you'll be on your way to creating a smart home that responds to a clap. It uses a microphone to listen out for the noise your hands make as they come together and triggers an action – in this case, switching an LED light on and off.

magpi.cc/clapper



Cool camera projects

21 Set up a security camera

Nobody wants a thief to scarper with their belongings, so it's always worth considering using a camera to keep a watchful eye on things (magpi.cc/smartcam). Grab the Raspberry Pi Camera Module v2 (or High Quality Camera) and you can begin building your own security surveillance network thanks to MotionEyeOS. What's more, this particular project does a mighty fine job in explaining each of the steps, looking at how you'll install the software, set up the Camera Module, and start adding extra cameras. As you may expect from the name of the operating system, this setup will detect motion and contact you if any issues arise. Clips can also be recorded when motion is detected.

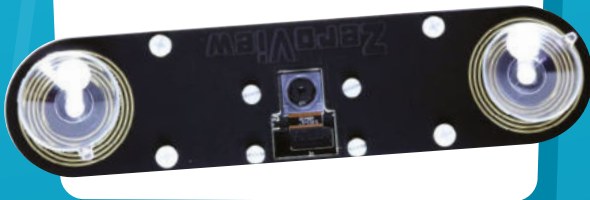
magpi.cc/pimylifecam



22 DRIVE AWAY WITH A DASH-CAM

Do you want to protect yourself from any motoring mishaps? Dash-cams can lower your insurance premiums, but they can be costly themselves. Buy the ZeroView add-on for less than a tenner and you can connect a Raspberry Pi Zero to a window. It includes a mount for your Raspberry Pi Camera Module.

magpi.cc/zeroview



23 FILM STOP-MOTION ANIMATION



Unleash your creativity and tap into a filmmaking technique dating back to the 19th century. Just think of something you'd like to film, set up the Camera Module or HQ Camera, and connect a button that will let you take a photo with Python. Then make tiny adjustments to your subject and snap away.

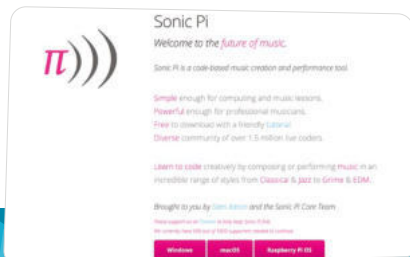
magpi.cc/pushbutton

Make music

24 CODE TUNES WITH SONIC PI

It's hard not to wax lyrical about Sonic Pi, the open-source platform for the live programming of music. All you need to start coding some tunes is the software built into Raspberry Pi OS, and either a pair of headphones or a speaker. Why not begin by programming a doorbell chime or trying your hand at a drum loop?

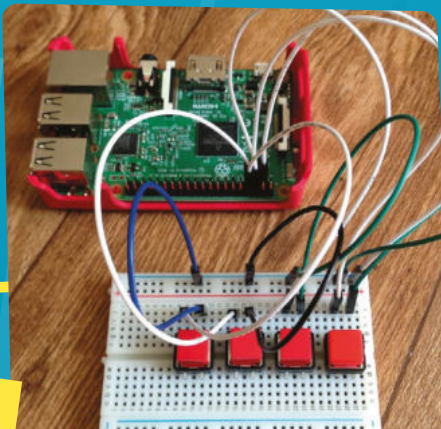
magpi.cc/sonicpi



25 MAKE A GPIO MUSIC BOX

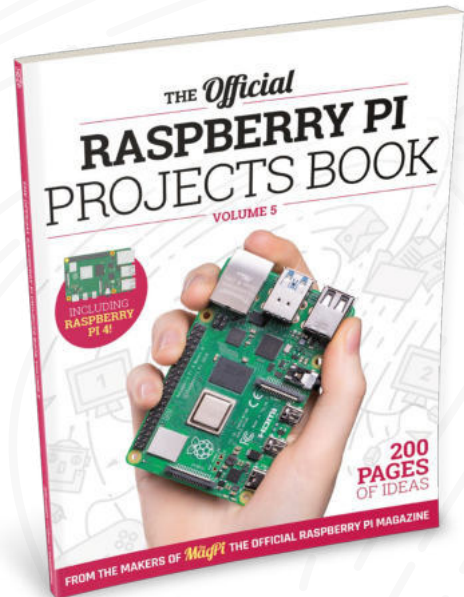
By wiring up buttons on Raspberry Pi's GPIO pins and running a small Python program, you can play different sounds when each one is pressed. This creates a simple musical instrument of sorts, although once you become familiar with how it all works, you can assign any sounds you wish.

magpi.cc/gpiomusicbox



THE *Official* RASPBERRY PI PROJECTS BOOK

VOLUME 5



200 PAGES OF RASPBERRY PI

- Use Raspberry Pi 4 as a desktop PC and 4K home media centre
- Learn how to code and make with our step-by-step tutorials
- Be inspired by incredible projects made by other people
- Find out about the top kits and accessories for your projects



Buy online: magpi.cc/store

MAGDOWNLOAD.ORG

FREE DOWNLOAD

LATEST MAGAZINES

HIGH QUALITY TRUE-PDF

MAGDOWNLOAD.ORG

DOSBox-X:

Boot Raspberry Pi to Windows 9x



KG Orphanides

KG enjoys making new computers pretend to be old ones, creating extremely niche games, and generating uncanny electronic screeching.

@KGOOrphanides

You'll Need

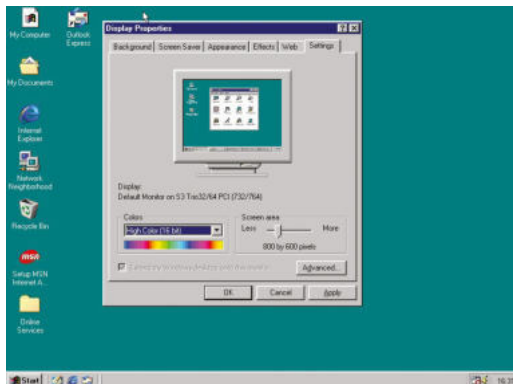
- ▶ DOSBox-X
magpi.cc/dosboxx
- ▶ Windows 98
magpi.cc/windows98
- ▶ DOSBox-X config files
- ▶ Windows 98 driver collection
magpi.cc/w98drivers
- ▶ USB floppy drive (optional)
- ▶ USB CD-ROM drive (optional)
- ▶ If you set Raspberry Pi to a 1280x720 resolution, then Windows will happily run at 800x600, but you may need lower settings for some software

We're going to create a hard disk image that will house a full install of Windows 98. Because it's a self-contained virtual drive, any software you want to install must be mounted as a floppy or CD image. We'll enable networking to provide access to the wider internet, although that's limited by lack of TLS 1.3 encryption support in older browsers.

This is going to be as close as you can get to the real Windows 98 experience on Raspberry Pi, driver dependencies and all. Although your disk image and install media will fit on an 8GB microSD card, you'll probably need a library of CD images, so a 32GB or larger card is strongly recommended.

01 Find a copy of Windows 98

Getting hold of Windows 98 is no longer easy as – unlike MS-DOS, or Windows 3.1 and 7 – Microsoft is no longer allowed to distribute Windows 95, 98, and Me through its Visual Studio



Subscriptions program due to an intellectual property claim by Sun.

That means you'll have to find a second-hand or unsold stock copy – eBay is a good bet for this – or rummage through your loft for old installation discs. The Windows product key is the really important bit here as the software is useless without it: check for key stickers on your old laptop or desktop PCs.

You'll also need a boot floppy to run most versions of the installation disc.

02 Image install media

It's easiest to install Windows 98 from images, so we'll want to copy both boot and install media. You can use another computer to image your discs and then copy them over, but if you have USB CD and floppy disk drives that play nicely with Raspberry Pi – not all do – then you can use `dd` in a Terminal window.

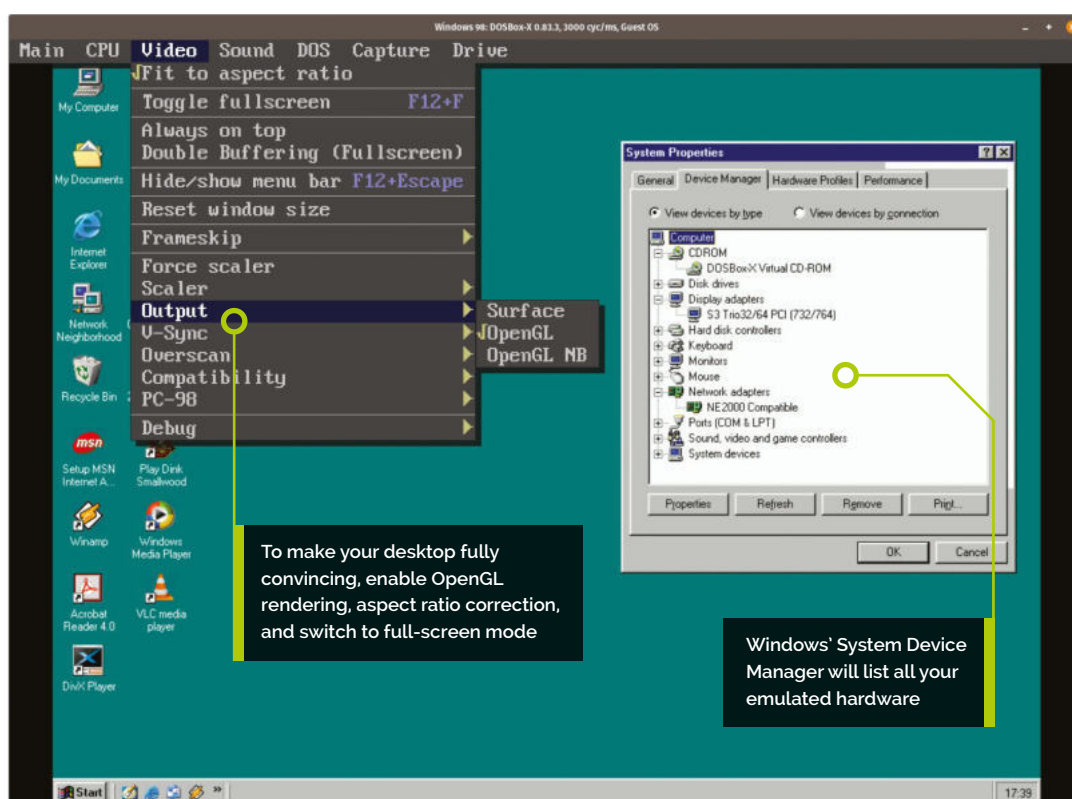
CD example:

```
dd if=/dev/sr0 of=win98.iso
```

Floppy example:

```
dd bs=512 count=2880 if=/dev/sda  
of=win98boot.img
```

In this tutorial, we've put all our media images in our home directory, but you may wish to create dedicated CD and floppy directories to house them in.



03 Install DOSBox-X

Open a Terminal window and enter:

```
sudo apt install automake libncurses-
dev nasm libSDL-net1.2-dev libpcap-dev
libfluidsynth-dev ffmpeg libavdevice58
libavformat-* libswscale-* libavcodec-*
git clone https://github.com/joncampbell1123/
dosbox-x.git
cd dosbox-x
./build
sudo make install
dosbox-x
```

DOSBox-X should open at its Z: prompt. Now exit, because we'll be using a custom config file for our Windows 98 shenanigans.

Before we get started in earnest, set Raspberry Pi's desktop resolution to 1280x720 under Preferences > Screen Configuration, accessible from the main menu: this improves full-screen performance under Windows 98.

04 Configure networking

Download our `win98_install.conf` and `win98.conf` files from magpi.cc/github and put them in your home directory. These files enable NE2000 networking, with NIC 1 selected. This is

correct if you're using Raspberry Pi 4's integrated wireless LAN. However, if you encounter any issues or wish to use a different internet connection, set `reallnic=list` in the DOSBox-X config file and check the Network Interfaces List that DOSBox-X will show in the Bash terminal window.

You'll need the PCAP library to give DOSBox-X network access.

```
sudo apt install pcap*
sudo setcap cap_net_raw+ep /usr/bin/dosbox-x
```

Check that this worked:

```
getcap /usr/bin/dosbox-x
```

Note that you'll have to rerun the `setcap` command whenever you upgrade DOSBox-X.

05 Create a Windows 98 disk image

From a Terminal, run:

```
dosbox-x -conf win98_install.conf
```

Now, from inside DOSBox-X:

```
imgmake win98hd.img -t hd_2gig -nfs
```

Top Tip

Shortcut to happiness

Use **LEFT-CTRL+F10** to release the mouse and **F12+F** to switch DOSBox-X in and out of full-screen mode.



Alert! Copyright

DOSBox is an emulator and we use it with open-source FreeDOS code. Be mindful of copyright when downloading image files, and only use proprietary software that you own and in accordance with the licence terms.

magpi.cc/dosboxlegal
magpi.cc/legaloms

Top Tip



Boot on the right foot

If you have a bootable Windows 98 ISO, you can mount its boot partition thus:

```
imgmount 0 -el-
torito D -t
floppy -fs none
```

We're creating a 2GB hard disk image here, as 4GB images (although supported on other systems) currently throw up virtual disk geometry errors on Raspberry Pi.

We've included, and partially commented out, lines to automatically mount and boot the images we'll use for installation in our `win98_install.conf` file. So once you've created this image, you can either uncomment them and restart DOSBox-X or enter them manually from at the DOS command prompt, as detailed in the following step.

06 Boot your Windows disc

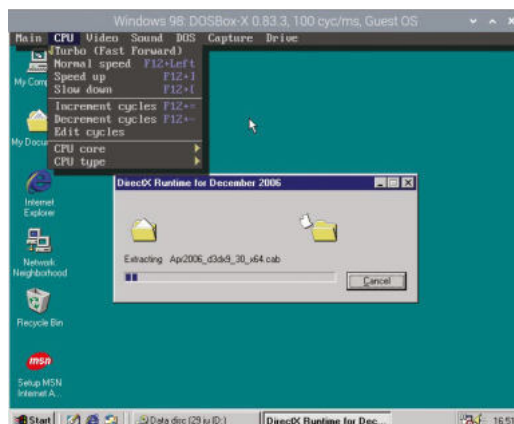
Enter the following at the DOS command prompt:

```
imgmount 2 win98hd.img -size 512,63,64,1023
-fs none
imgmount a win98boot.img -t floppy
imgmount d win98.iso -t iso -fs none -ide
2m
boot A:
```

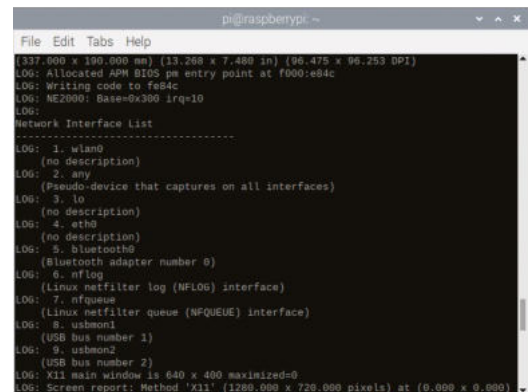
Start Windows 98 Setup from CD-ROM, press **ENTER** to continue with setup as prompted, opt to configure unallocated disk space and enable large disk support, then press **ENTER** to restart.

DOSBox-X will restart its guest system and you – or your config file – should rerun the above mount and boot commands again.

Choose to start setup from CD-ROM again and the installer will format your blank disk image. Press **ENTER** when prompted to carry out the pre-installation check.



▲ Enable Turbo mode to speed up lengthy installations – just remember to switch it back to Normal speed when you're done



▲ Set `realnic=list` in your config file and DOSBox-X spits out a numbered list of network interfaces. Set `realnic` to the number you want to use.

07 Install-time configuration

Agree to the EULA, then enter your Windows 98 Product Key. Click Next to accept the default **C:\WINDOWS** install directory. A 'Typical' install from the next set of options works for most users; select 'Custom' if using a non-Latin character set.

Enter your name when prompted and move on to component selection, where you'll find those additional character set options. Name your system, enter your network name, then set your keyboard, language, and location.

Skip creating a startup disk by clicking Next, then Cancel when you're prompted to insert a disc. The installer is finally ready to copy Windows 98's files. Enable Turbo Mode from the DOSBox-X CPU menu to speed this time-consuming process along. The guest system will reboot when finished.

08 Windows driver initialisation

At the DOSBox-X Z: prompt, type:

```
boot win98hd.img
```

This boots into Windows 98 and, along with mount commands for the hard disk and a couple of ISO images, is already in our supplied `win98.conf` file, which you can use by launching DOSBox-X with:

```
dosbox-x -conf win98.conf
```

Windows 98 will take several minutes to initialise its drivers and may reboot during this process. Set your time zone and network settings – the virtual network card's IRQ should be set to 10 – when prompted. Setup will continue and the system will reboot.

At its first real boot, Windows will ask you to set a network password that will also be used for your system and then install more drivers from the CD.



▲ Raspberry Pi can struggle with modern USB disc drives, but simple CD-ROM readers like the Teac CD-210PU work well

09 Copy your install media

You should have functional sound, Voodoo graphics, and networking. Tell Windows that you're using a LAN connection. If anything fails to install correctly, use the Add New Hardware wizard and Device Manager to find and configure the relevant items.

To make future driver installation easier, copy the **/win98** folder from the install CD image to your virtual hard disk using Windows File Manager. You can direct Windows to default to checking this directory in place of the CD by opening regedit, navigating to **HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Setup** and changing **SourcePath** to the location where you copied the files. That's **SourcePath=C:\\WIN98** in our case.

10 Update your drivers

DOSBox-X allows you to mount multiple disc images using its **imgmount** command, and switch between them by pressing **F12+LEFT-CTRL+C**. Our config file includes an example and we've created an ISO image full of drivers, demos, and free software to get you started at magpi.cc/github.

Install, in this order: 7-zip, Windows Installer 2.0 (instmsi), Internet Explorer 6, dotnet 1, dotnet 2, the visual C++ redistributable, the K-Lite Codec Pack, Windows Media Player (MPClassic), and the Sound Blaster 16 driver.

Enable Turbo Mode to speed up the slowest installers, but switch to Normal speed before you type anything.

win98.conf

► Language: DOSBox-X config file

DOWNLOAD
THE FULL CODE:



magpi.cc/github

```
001. [sd1]
002. # set fullscreen true if you want to launch in fullscreen mode
003. fullscreen=false
004. autolock=true
005. fullresolution = desktop
006. output = opengl
007.
008. [dosbox]
009. title=Windows 98
010. memsize=512
011. cycles=60000
012. vesa modelist width limit=0
013. vesa modelist height limit=0
014.
015. [render]
016. # aspect ratio correction
017. aspect=true
018.
019. [cpu]
020. # pentium_mmx is recommended for Windows 98 installs but causes
    audio distortion in some games: switch cputype to auto if this
    occurs
021. cputype=pentium_mmx
022. core=normal
023.
024. [sblaster]
025. sbtype=sb16vibra
026. irq=5
027. dma=1
028. hdma=5
029.
030. [ne2000]
031. # If you want networking in Windows, set ne2000=true.
032. # This also requires that you set realnic= to a suitable value
    for your PC - that should be 1 for Raspberry Pi 4 wlan
033. ne2000=true
034. nicirq=10
035. realnic=1
036.
037. [fdc, primary]
038. int13fakev86io=true
039.
040. [ide, primary]
041. int13fakeio=true
042. int13fakev86io=true
043.
044. [ide, secondary]
045. int13fakeio=true
046. int13fakev86io=true
047. cd-rom insertion delay=4000
048.
049. [render]
050. scaler=none
051.
052. [autoexec]
053. imgmount 2 win98hd.img -size 512,63,130,1023 -fs none
054. # imgmount a win98boot.img -t floppy
055. imgmount d win98.iso Win98Soft.iso
056.
057. boot win98hd.img
```


win98_install.conf

► Language: DOSBox-X config file

```
001. [sdl]
002. autolock=true
003. fullresolution = desktop
004. output = opengl
005.
006. [dosbox]
007. title=Windows 98
008. memsize=64
009. cycles=60000
010. vesa modelist width limit=0
011. vesa modelist height limit=0
012.
013. [render]
014. # aspect ratio correction
015. aspect = true
016.
017. [cpu]
018. cputype=pentium_mmx
019. core=normal
020.
021. [sbblaster]
022. sbtype=sb16vibra
023. irq=5
024.
025. [ne2000]
026. # If you want networking in Windows, set ne2000=true.
027. # This also requires that you set realnic= to a suitable value
    for your PC - that should be 1 for Raspberry Pi 4 wlan
028. ne2000=true
029. nicirq=10
030. realnic=1
031.
032. [fdc, primary]
033. int13fakev86io=true
034.
035. [ide, primary]
036. int13fakeio=true
037. int13fakev86io=true
038.
039. [ide, secondary]
040. int13fakeio=true
041. int13fakev86io=true
042. cd-rom insertion delay=4000
043.
044. [render]
045. scaler=none
046.
047. [autoexec]
048. imgmount 2 win98hd.img -size 512,63,130,1023 -fs none
049. imgmount d win98.iso
050. imgmount a win98boot.img -t floppy
051. # If you have a bootable Win98 disc replace the above line with:
052. # imgmount 0 -el-torito D -t floppy -fs none
053. boot A:
```

11 Install some software

Now you have a Windows 98 PC, you'll want something to run on it. Our software disc includes some freeware and shareware applications, including Dink Smallwood and the first episode of Doom.

Online auctions can supplement your collection and some publishers have made parts of their back catalogue available for free. That includes Dynamix's Tribes 2 mech combat sim (tribesnext.com), three of EA's Command & Conquer RTS series (cncnet.org/buy), and some Battlecruiser space sims (3000ad.com/download).

Innoextract (constexpr.org/innoextract) can extract Windows 98 compatible files from GOG packages, KernelEx (kernelex.sourceforge.net) runs Windows XP software, and Brasero (`sudo apt install brasero`) can burn downloaded software to disc images for easy mounting.

“To really feel like a Windows 98 PC, you'll want it to boot directly to the OS”

12 Boot to Windows

If you want Raspberry Pi to really feel like a Windows 98 PC, you'll want it to boot directly to DOSBox-X's guest operating system. Open your config file, set `fullscreen=true`, and add any disc images you plan on using to the `[autoexec]` section. Also make a note of the shortcuts you'll need to swap CD and floppy images.

```
mkdir /home/pi/.config/autostart
mousepad /home/pi/.config/autostart/dosbox.
desktop
```

Add the following to the new text file and save it:

```
[Desktop Entry]
Type=Application
Name=DOSBox
Exec=/usr/bin/dosbox-x
```

Copy your config file to DOSBox-X's default config file location:

```
cp /home/pi/win98/conf /home/pi/.config/
dosbox-x/dosbox-x-0.83.3.conf
```

Now reboot Raspberry Pi into Windows. 

— **Uni**versal & **Uni**que —



UniPiCase

for Raspberry Pi 4

Perfect option for your DIY or high-volume commercial projects



Fits all kinds of HATs
Ideal PoE HAT case

Use with HAT or Pi alone
Simple and professional

Rapid, tool-free assembly
GPIO cable pass-through

Two wall mount options
Well ventilated



www.UniPiCase.com

Designed and manufactured in North America

High Quality Camera: Stop-motion and selfies

Add a shutter button to take pictures and use it in projects

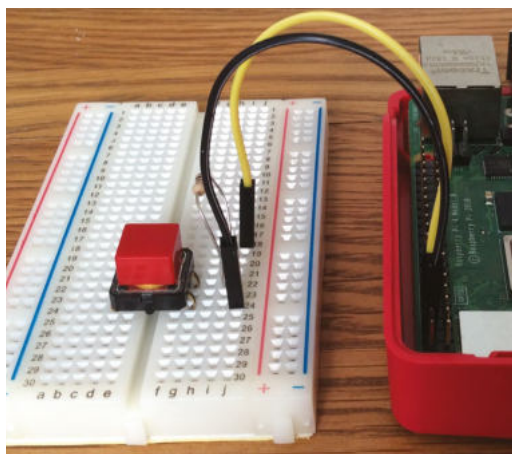
Have you been reading the last few instalments of this series and thinking you'd like to take a picture with a Raspberry Pi camera with less hassle? In this tutorial we'll show you how to take a photo with a click of a button, just like a real camera. This could be useful for many projects (for example, time-lapse photography), but in this tutorial we are focusing on stop-motion animation. We also show how to create your own selfie stick!

01 Wire up the button

If you haven't already switched your Raspberry Pi off, do so now. Next, connect the button to Raspberry Pi via a jumper wire, as shown in **Figure 1**. One side of the button will be connected to ground (GND); the other is

You'll Need

- Camera Module / HQ Camera
- Push-button
- Breadboard (optional)
- Jumper wires
- Raspberry Pi case with a hole for the camera cable (for selfie stick)
- Long wires (for selfie stick)
- A stick, slim metal pole etc. (for selfie stick)



▲ You can use a breadboard for a small button, or connect your jumper wires directly to the pins on a bigger one

connected, via a resistor, to the GPIO 14 pin. We used a breadboard for our stop-motion animation project, but you could wire the button directly to the pins (as you'll be doing for the selfie stick later).

02 Install picamera

That's all the hardware done. Now it's time for the software. If you haven't done so already, you'll need to install the picamera library. In a Terminal window, enter:

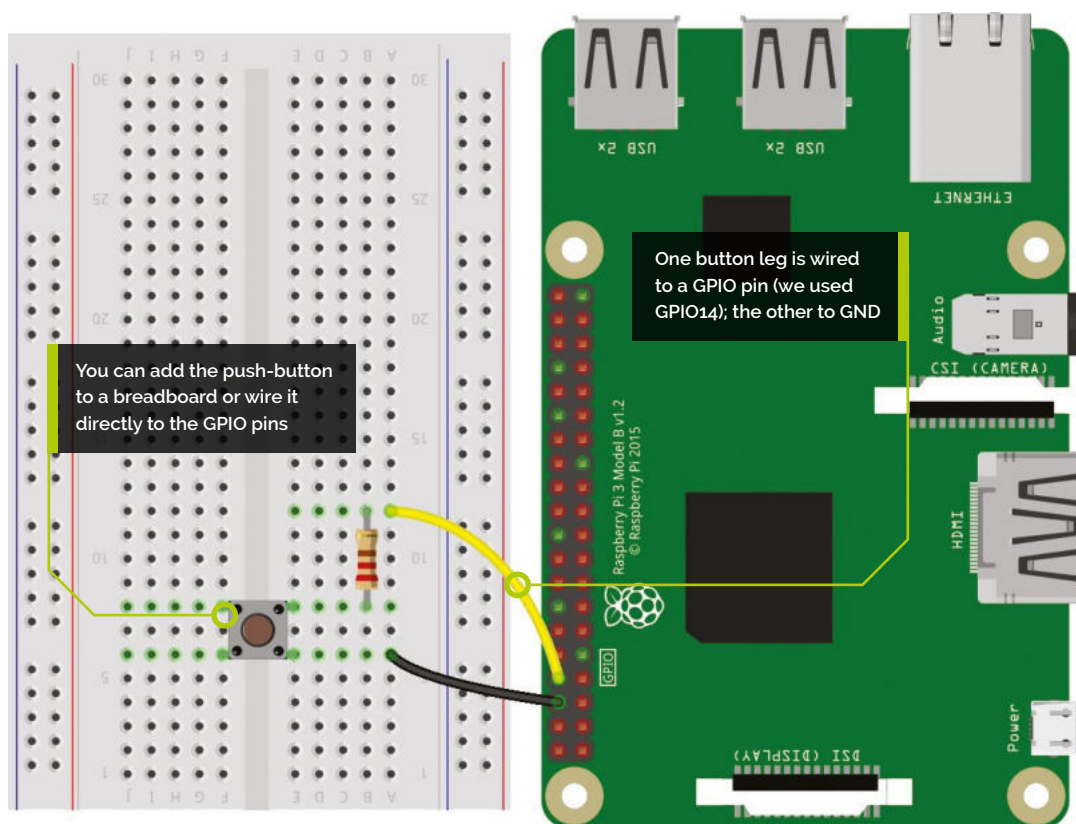
```
sudo apt-get update
sudo apt-get install python-picamera
python3-picamera
```

If for some reason you don't have GPIO Zero already installed (it has come pre-installed in Raspberry Pi OS for some time), do so with:

```
sudo apt-get install python-gpiozero
python3-gpiozero
```

03 Stop-motion software

Because we're focusing on stop-motion for our first project, we're using the camera's preview mode so that we can set up our shot before we take it, to ensure everything is in the frame. Then, only when the button is pressed do we save an image file. Each image file will have a different name based on the date and time at which it is taken. This makes it easy to assemble all the images from the shoot for post-processing.



◀ **Figure 1** Connect the button

The wonderful GPIO Zero library is used to capture the button activity; we simply define a function that is run whenever the button is pressed. This function uses the picamera Python library which allows us to control the camera through code, making all the normal command-line operations available.

Download or type up the code from **listing1.py** and either run it through Thonny or the command line. To quit the program, press **CTRL+C**.

04 Make the video

The final step is to assemble the video, using the FFmpeg tool (`sudo apt-get install ffmpeg`). Run the following command:

```
sudo ffmpeg -i /var/www/html/frame%04d.jpg
-cr 4 -b:v 10M /var/www/html/video.webm
```

When the rendering process has finished, you'll be able to view the video in your browser. The default frame rate is 25 fps. This compresses three hours of images taken at ten-second intervals to about 40 seconds of video. You can adjust this

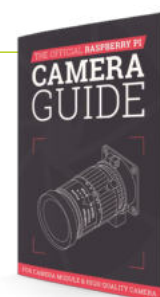
“ The picamera Python library allows us to control the camera through code ”

with the `-framerate` command-line option. The bitrate (`-b`) has been set high, and the Constant Rate Factor (`-crf`) has been kept low, to produce a good-quality video.

05 Other variations

You should be able to use this code as a template to create a program for whatever photography project you have in mind. For example, you could alter the code so that the camera takes continuous photos while the button is held down. Or you could add extra buttons to make a variety of photography modes available.

With this sort of build, you can also start thinking about building a complete, portable, wirelessly connected Raspberry Pi camera. For this, you can use a case into which you can fit a portable mobile



The Official
Raspberry Pi
Camera Guide

For further tutorials on how to use your HQ Camera or Camera Module, take a look at our new book, *The Official Raspberry Pi Camera Guide*. Its 132 pages are packed with essential info and a range of exciting projects. magpi.cc/cameraguide

phone battery charger, along with a screen to attach to Raspberry Pi. See issue 94's High Quality Camera case hack tutorial for inspiration (magpi.cc/94).

With a bit of modification of the code, you can have it always show the preview of the camera on the screen. Want to record video? More modification of the code will allow for video capturing. The only issue you might have with both of these projects is the lack of a flash or built-in light source, so a well-lit subject would be essential.

06 Selfie stick

Next, we'll look at making a selfie stick – we've used the original, smaller Camera Module for ours. A lot of people roll their eyes and complain about vanity when it comes to the art of

the selfie, but we all know it's nothing like that. New outfit? New glasses? Eyeliner wings perfectly symmetrical today? Why not chronicle it? It's a great confidence boost.

Our Raspberry Pi-powered selfie stick will use a similar hardware and software setup to the stop-motion animation project. As before, we're wiring up a push-button to GPIO 14 and GND pins on Raspberry Pi, but this time we need to attach the jumpers to longer wires to put the button at the end of the 'stick' – we used a kitchen spatula, but anything long will do.

Your Raspberry Pi needs to be near to the camera (unless you've got an extra-long ribbon cable). Attach Raspberry Pi in a case to one end of the stick with whatever means you see fit (glue, adhesive putty, string, etc.) and then attach the button.

“ Attach jumpers to longer wires to put the button at the end of the 'stick' ”

07 Add the code

Since the principle is the same – pressing a button to take a photo – we can use the same code, **listing1.py**, as for the stop-motion project. This time we don't need the camera preview, so you can comment out the line `pc.start_preview()` if you like, by adding a `#` to the start of it.

Try running the code. Pressing the button will take a photo, but you'll need to practise your aim so you can get yourself in the frame. As before, we add a timestamp to each picture, which helps to organise your pictures later and also results in a slight pause in the code, which at least means you won't take too many pictures with a slip of the button. ■



▲ Turn an official Raspberry Pi case into the ultimate hackable camera rig

listing1.py

► Language: Python 3

DOWNLOAD
THE FULL CODE:

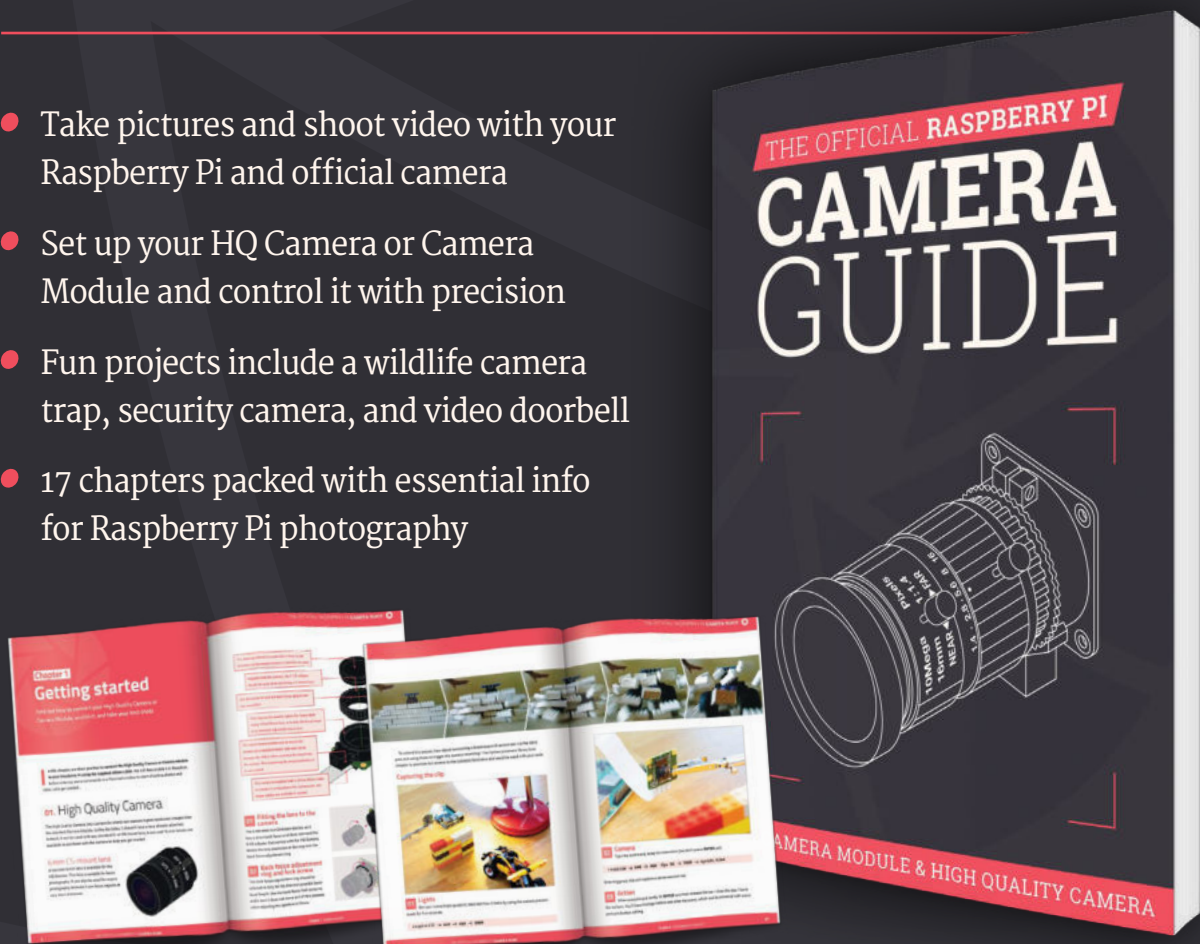


magpi.cc/github

```
001. #importing the necessary modules
002. from datetime import datetime
003. from gpiozero import Button
004. import picamera
005. import time
006.
007. b=Button(14)
008. pc=picamera.PiCamera()
009. running = True
010. #pc.resolution = (1024, 768)
011. #use this to set the resolution if you dislike the default
    values
012. timestamp=datetime.now()
013. def picture():
014.     pc.capture('pic'+str(timestamp)+'.jpg') #taking the
        picture
015.
016. pc.start_preview() #running the preview
017. b.when_pressed=picture
018. try:
019.     while running:
020.         print('Active')#displaying 'active' to the shell
021.         time.sleep(1)
022. #we detect Ctrl+C then quit the program
023. except KeyboardInterrupt:
024.     pc.stop_preview()
025.     running = False
```

THE OFFICIAL RASPBERRY PI CAMERA GUIDE

- Take pictures and shoot video with your Raspberry Pi and official camera
- Set up your HQ Camera or Camera Module and control it with precision
- Fun projects include a wildlife camera trap, security camera, and video doorbell
- 17 chapters packed with essential info for Raspberry Pi photography



Buy online: magpi.cc/cameraguide

Build the ultimate home music system



PJ Evans

PJ is a writer, software engineer, and lover of open-source software. It's his turn to choose the next track, thank you very much.

@mrpjevans

Want your own whole-house audio without the price tag? In this three-part series, we create a digital jukebox using open-source software

The ability to play music throughout your home, in perfect sync and controllable from your phone, has been around for a few years now. Many solutions are available, from simple Bluetooth speakers to high-end systems like Sonos. Often the price can be off-putting and there has been recent controversy over the closed nature of these systems. Can we build a whole-house audio system using nothing but open-source technology and our favourite little computer? This month, we'll start by building a dedicated music-playback station that can produce audiophile-level sound and web-based control.

01 Hardware assembly

The JustBoom DAC (digital-to-analogue converter) HAT provides high-quality audio output for our project. It also blocks the GPIO header, which we need to power the touchscreen. Luckily, the HAT has a GPIO breakout, so we soldered a two-pin header to the 5V and GND for the touchscreen so the whole system can be

powered by a single connector. You can also use two power supplies if you prefer. Mount the DAC HAT on the GPIO pins and secure. We used 20 mm 2.5M screws to secure the HAT to the computer and screen controller standoffs. Connect the 5V and GND pins for the touchscreen to their partners on the DAC HAT.

02 Prepare Raspberry Pi OS

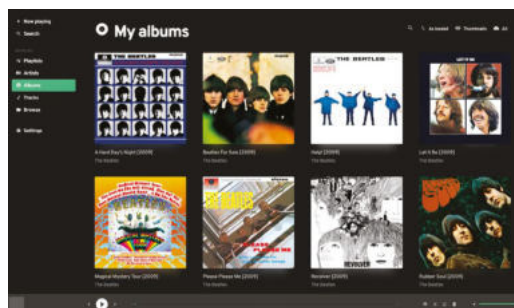
The case prevents access to the microSD card, so make sure you've got a basic setup of Raspberry Pi OS before assembling everything. For this project we're going to be using Chromium, so we need the 'standard' OS image with Desktop (but we don't need the Recommended Software version). Burn this image with your preferred method and make sure you're connected to the network with SSH enabled, or hook up a mouse and keyboard if you prefer. Finally, as ever, make sure you're running the latest version of everything with `sudo apt update && sudo apt upgrade`.

03 Mount into the case

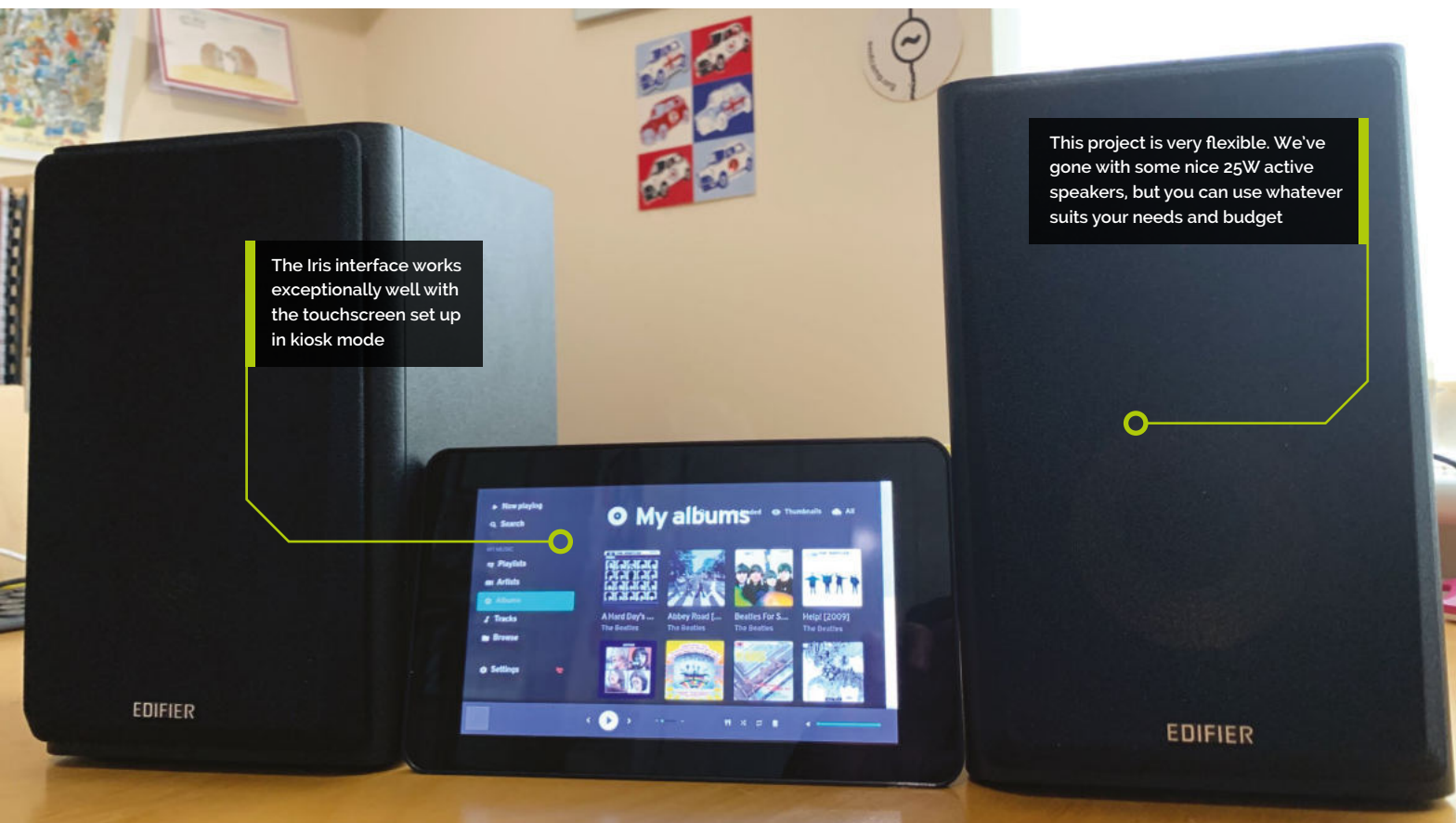
The Raspberry Pi 4B case used here will not accept the assembly with the DAC HAT. If you're not using a HAT, you can skip this step. Otherwise, carefully remove the two wall-mounting screw holes and then line up the HAT with the top ventilation spacers and remove any that are blocking the two phono sockets and the 3.5 mm socket from sitting flush in the case. Measure twice and cut once! The plastic is soft, so it's fairly easy to do. Once the assembly is sitting comfortably in the case, you can screw in the touchscreen and it should now sit up securely.

You'll Need

- JustBoom DAC HAT magpi.cc/justboomdac
- Official Raspberry Pi 7" Touchscreen magpi.cc/touch
- Touchscreen Case magpi.cc/touchcase
- Active speakers, e.g. magpi.cc/r1000t4



▲ Iris is a touchscreen-friendly, web-based interface for Mopidy. It supports add-ons for popular platforms such as Spotify and Soundcloud



04 Enable DAC output

No sound will come from the DAC HAT by default. Luckily, the required drivers are already installed and waiting to be enabled. From the command line, run the following:

```
sudo nano /boot/config.txt
```

Find the line that reads:

```
dtoverlay=audio=on
```

Remove the line (or add a # to the front to comment it out), then add these lines:

```
dtoverlay=audio=off
dtoverlay=justboom-dac
lcd_rotate=2
```

That last line flips the screen if you've found, as we did, that the touchscreen display is now upside down. Reboot your Raspberry Pi computer and from the command line, test audio output:

```
aplay /usr/share/sounds/alsa/Front_Center.wav
```

You should hear a calm voice say 'front centre'.

“ Luckily, the required drivers are already installed and waiting to be enabled ”

05 Install Mopidy

Mopidy is a headless music server. This means it has no user interface of its own, so we'll add one later. To install Mopidy, we need to add its repository to APT. Run the following commands one-by-one:

```
wget -q -O - https://apt.mopidy.com/mopidy.gpg | sudo apt-key add -
sudo wget -q -O /etc/apt/sources.list.d/mopidy.list https://apt.mopidy.com/buster.list
sudo apt update
```

Now install Mopidy and set its user's permissions:

```
sudo apt install mopidy
sudo adduser mopidy video
```

Set Mopidy to run as a background service on boot:

```
sudo systemctl enable mopidy
```

Top Tip

New album blues

New music not appearing? Go to Settings and then click 'Run Local Scan'. If tracks still don't appear, click 'Reset Cache' on the same screen.

Top Tip

Keep it simple

Change things to suit your budget. You can run Mopidy on a full-size Raspberry Pi with the built-in headphone socket.

06 Configure and test Mopidy

Before Mopidy can be run, we need to configure a few things. The default settings are sensible, but if you want to access the web interface (once installed) from another device, we'll need to allow that.

```
sudo nano /etc/mopidy/mopidy.conf
```

After the comments block, add this:

```
[http]
hostname = 0.0.0.0

[audio]
output = alsasink
```

Save the file with **CTRL+X** and then start Mopidy:

```
sudo systemctl enable mopidy
```

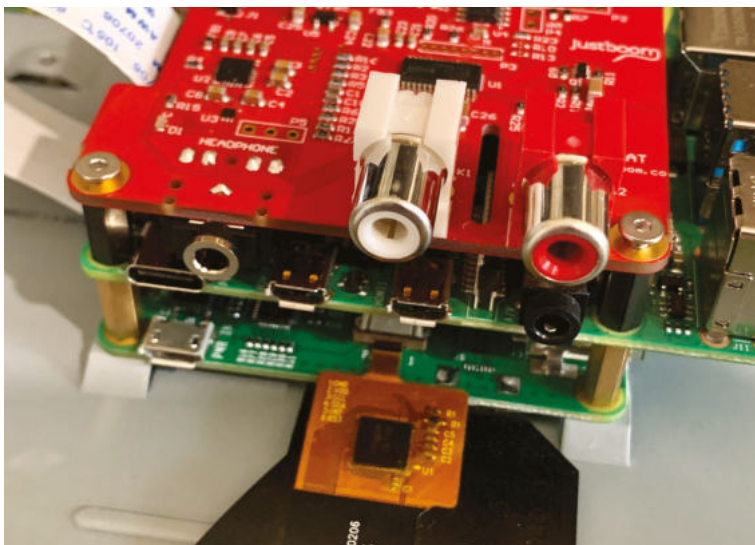
In a browser, go to **http://<hostname>:6680/** (replace <hostname> with your Raspberry Pi's name – 'raspberrypi.local' by default). If you get a welcome page displayed, all is working well.

07 Install the UI & file support

Next, we're going to install Iris, a user interface, and support for playing music stored on the local system. Run these commands:

```
sudo pip3 install Mopidy-Iris
```

▼ The DAC HAT provides high-resolution output at line level. You can swap this with an amplifier HAT and use passive speakers instead



```
sudo pip3 install Mopidy-Local
sudo sh -c 'echo "mopidy ALL=NOPASSWD:
/usr/local/lib/python3.7/dist-packages/
mopidy_iris/system.sh" >> /etc/sudoers'
sudo systemctl restart mopidy
```

This installs the Iris and Local modules, then gives Iris permission to run Mopidy commands as the superuser. Finally, we restart Mopidy to pick up the changes. In a web browser, go to **http://<hostname>:6680/iris/** to see the interface.

08 Just add music

Mopidy supports a wide range of formats, including MP3 and FLAC. Whatever you decide to use, create a space for your music and upload some files there. You'll find a **Music** folder already exists in your home directory. We need to tell Mopidy where to look for your files. Edit the Mopidy configuration file:

```
sudo nano /etc/mopidy/mopidy.conf
```

Add append the following:

```
[local]
media_dir = /home/pi/Music
```

Restart Mopidy to pick up the configuration changes and then initiate the first music scan:

```
sudo mopidyctl local scan
```

Now in Iris, you should be able to see your albums appear. Try playing one!



▲ We've modified the case and removed the back so everything fits. The screen takes its 5V feed from the GPIO breakout on the DAC HAT

09 Kiosk mode

To get Iris running on the touchscreen, we can make Chromium run in ‘kiosk’ mode so that no other elements of Raspberry Pi OS Desktop can be seen. The first step is to install ‘unclutter’, a handy app for hiding the mouse pointer when not in use.

```
sudo apt install unclutter
```

Now we’ll create a script to launch Chromium with all the correct settings:

```
nano ~/kiosk.sh
```

Add the code shown in the listing. When run, this will enable cursor hiding, prevent some Chromium error messages from being displayed, and then start Chromium in kiosk mode, going directly to the Iris website in full-screen.

10 Implement the kiosk script

Before we can use the script, it needs to be flagged as an executable:

```
chmod +x ~/kiosk.sh
```

Now try it out by running `~/kiosk.sh`. If you’re entering commands remotely via SSH, use this command instead:

```
DISPLAY=:0 ~/kiosk.sh
```

All being well, Iris should start full-screen and be controllable by touch. We recommend going into settings and enabling ‘Wide scrollbars’ to make it easier to control with fingers. You can also choose between light and dark themes, the darker theme being a little kinder on the eyes when used with the touchscreen.

11 Run on boot

The final step for our music system is to make it run on boot and go straight into kiosk mode. We’ll ask the Desktop itself to do this. Enter the following in the command line:

```
sudo nano /etc/xdg/lxsession/LXDE-pi/autostart
```

Add this line, including the ‘@’ sign:

kiosk.sh

► Language: **Bash/Shell**

**DOWNLOAD
THE FULL CODE:**



magpi.cc/iriskiosk

```
001. #!/bin/bash
002. unclutter -idle 0.5 -root &
003. sed -i 's/"exited_cleanly":false/"exited_cleanly":true/'
    /home/pi/.config/chromium/Default/Preferences
004. sed -i 's/"exit_type":"Crashed"/"exit_type":"Normal"/'
    /home/pi/.config/chromium/Default/Preferences
005. /usr/bin/chromium-browser --noerrdialogs
    --disable-infobars --kiosk http://127.0.0.1:6680/iris &
```

```
@/home/pi/kiosk.sh
```

Save the file and reboot your system. The desktop will appear and then Iris will boot full-screen. Your touchscreen music system is now ready for use. You can control everything from either the touchscreen, a web browser on any device, or your smartphone.

12 Load up the music


If you want to add some audiophile-grade music to your collection, you can use your Raspberry Pi computer to rip CDs to FLAC, a lossless compression system. The files are large, but no music detail is lost. A very easy-to-use ripper is ‘A Better CD Encoder’ aka ‘abcde’.

```
sudo apt install abcde
```

Hook up a CD drive to the USB port, insert a CD, and enter this:

```
abcde -o flac
```

Abcde will guide you through the whole process, including automatically naming the tracks. Place the resulting directory in your **Music** directory and go to Settings in Iris to initiate a rescan.

Mopidy, with a little help from its friends, can play music from many different services, including Spotify. Next month we’ll explore how. 

We’d like to acknowledge PiMyLifeUp for their excellent tutorial on kiosk mode: magpi.cc/pikiosk.

Remote access your Raspberry Pi securely



MAKER

Lucy Hattersley

Lucy is editor of *The MagPi* and has worked remotely for years. She has a Raspberry Pi ready to be connected to from anywhere in the world.

@lucyhattersley

Use a secure service to gain remote access to your Raspberry Pi

The small footprint and low energy demands of Raspberry Pi make it ideal for remote deployment, and accessing Raspberry Pi from another computer is a common user request.

There are many different solutions for remote access to Raspberry Pi. The most common is to use one of two techniques: SSH (magpi.cc/ssh) and VNC (magpi.cc/vnc).

SSH and VNC involve opening a port on Raspberry Pi (VNC uses port 5900+N and SSH uses port 22). This potentially exposes your Raspberry Pi. Hackers actively look for Raspberry Pi devices with these open ports and default passwords. This is why you must change your password when turning on VNC (magpi.cc/security).

Some people take things further and use a third-party service to help protect their Raspberry

Pi. In this tutorial, we're taking a look at remote.it (magpi.cc/remotedit) to set up a work network using Raspberry Pi as a remote access gateway.

With remote.it, you can access a Raspberry Pi remotely without port forwarding. You do this by accessing the remote.it server as a proxy, or by using remote.it's software to form a peer-to-peer network. [Remote.it](https://remote.it) claims this is a safer way to set up a gateway than a traditional VPN (magpi.cc/vpnhackers).

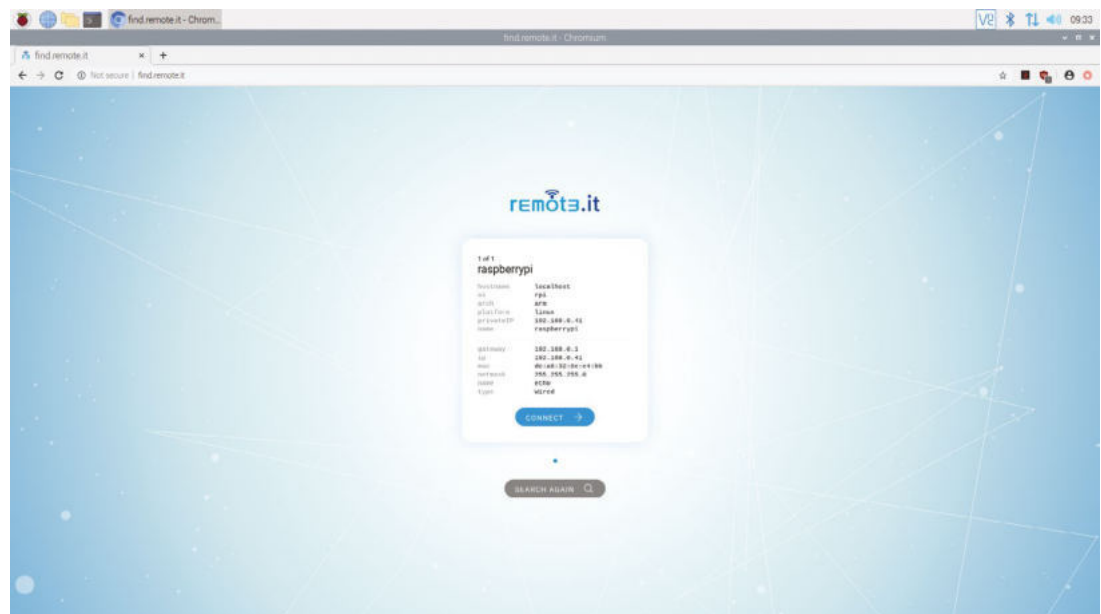
01 Create an account

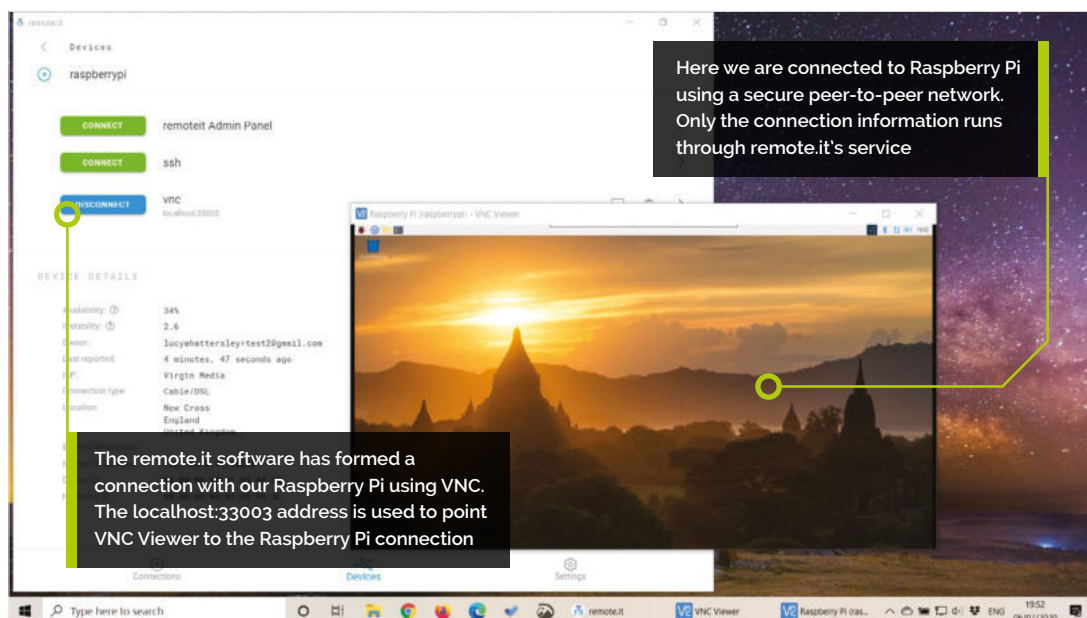
To connect to your Raspberry Pi remotely you'll need to set up an account. Open the web browser and head to <https://remote.it>. Click on 'Sign up' and enter your email address. You will

You'll Need

- Raspberry Pi
- remote.it account
- Raspberry Pi OS
- VNC Viewer
- magpi.cc/vncviewer

- Use the [http://find.remote.it](https://find.remote.it) webpage to locate your Raspberry Pi on the network and add it to your remote.it account





be emailed a verification code. Enter this code and pick (and confirm) a password. Make sure you choose a safe password (read this [How To Geek](https://www.howtogeek.com/2011/10/25/strong-password/) article: magpi.cc/strongpasswd).

02 Turn on SSH and VNC

To connect remotely, SSH and VNC must be activated on Raspberry Pi.

Click on the main menu and choose Preferences > Raspberry Pi Configuration. Choose the Interfaces tab and set both SSH and VNC to Enabled.

The VNC icon appears in the menu bar. Make sure you are not using the default password. (Choose the System tab in Raspberry Pi Configuration and click Change Password.) See the Raspberry Pi documentation 'Securing your Raspberry Pi' article for more information on keeping your computer safe: magpi.cc/security.

03 Update and install

Make sure your Raspberry Pi is connected to the internet (using wireless LAN or an Ethernet cable). Open a Terminal window and enter the following commands:

```
sudo apt update
sudo apt install remoteit
```

When the packages are installed, the Terminal outputs the configuration information:

```
Continue your device configuration at
http://find.remote.it
or http://raspberrypi.local:29999
or http://192.168.0.41:29999
```

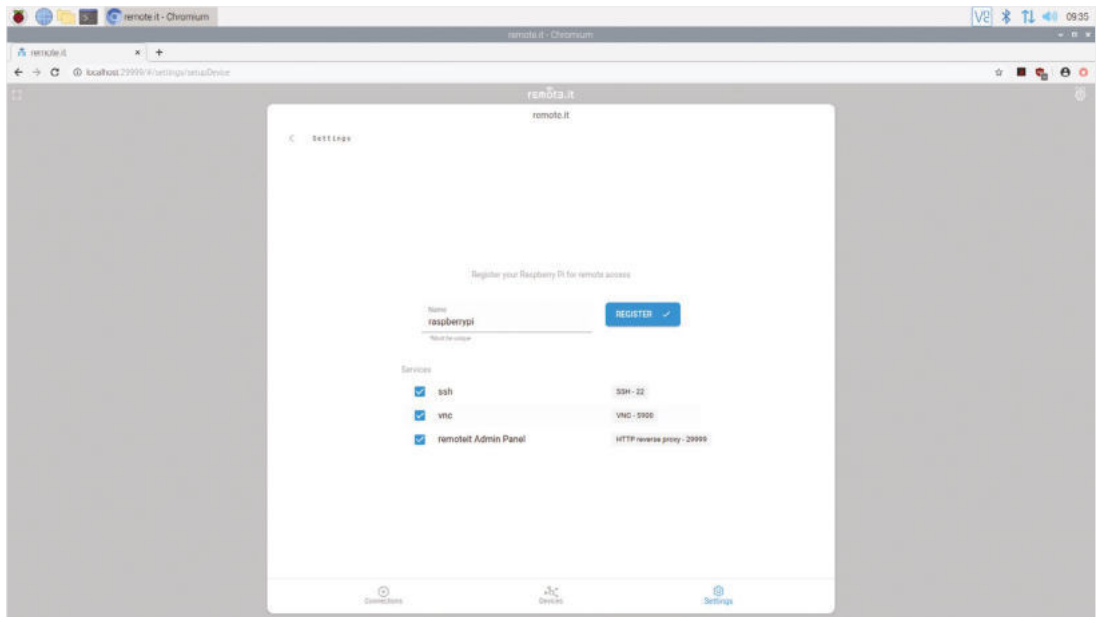
The IP (internet protocol) address will be unique to your network.

“ With remote.it, you can access a Raspberry Pi remotely without port forwarding ”

04 Register the Raspberry Pi

Open the web browser on your Raspberry Pi and navigate to <http://find.remote.it>. The window will search for devices running remote.it software (including your Raspberry Pi). When your Raspberry Pi is found, the screen will display 'Unregistered device' along with the gateway details and IP address. (If this approach does not work, try one of the other two URLs listed in the configuration information in Step 3). Click on Register to add your Raspberry Pi to remote.it. You will be asked to enter the Email and Password for your remote.it account (as created in Step 1).

► Register the SSH and VNC connections from Raspberry Pi OS with your remote.it account



Top Tip

Use a prepared OS image

Remote.it offers a preconfigured software image that has all the required software and settings installed. remote.it/downloads

The Name field displays 'raspberrypi' (you can change the name to something else if you prefer). The Services window displays the three default services: ssh, vnc, and remoteit Admin Panel. Click Register again to add Raspberry Pi to your remote.it account.

05 Start a connection

You now connect to Raspberry Pi via remote.it. There are two approaches: via a web service, or using remote.it's dedicated software.

First, let's look at the web service (magpi.cc/remoteweb). This provides a link to a proxy connection, and all data passes through one of remote.it's proxy servers.

Open a web browser on your other computer and navigate to app.remote.it (enter your remote.it login credentials).

Click Connect and choose 'Raspberry Pi - VNC'. The 'VNC - Connection' window appears, displaying host and port information.

06 Connect via VNC Viewer

Open your VNC software (we're using VNC Viewer: magpi.cc/vncviewer) and choose File > New connection. Cut and paste the combined proxy and port information into the VNC server field. Ours looks like this: **proxy50.rt3.io:31249**.

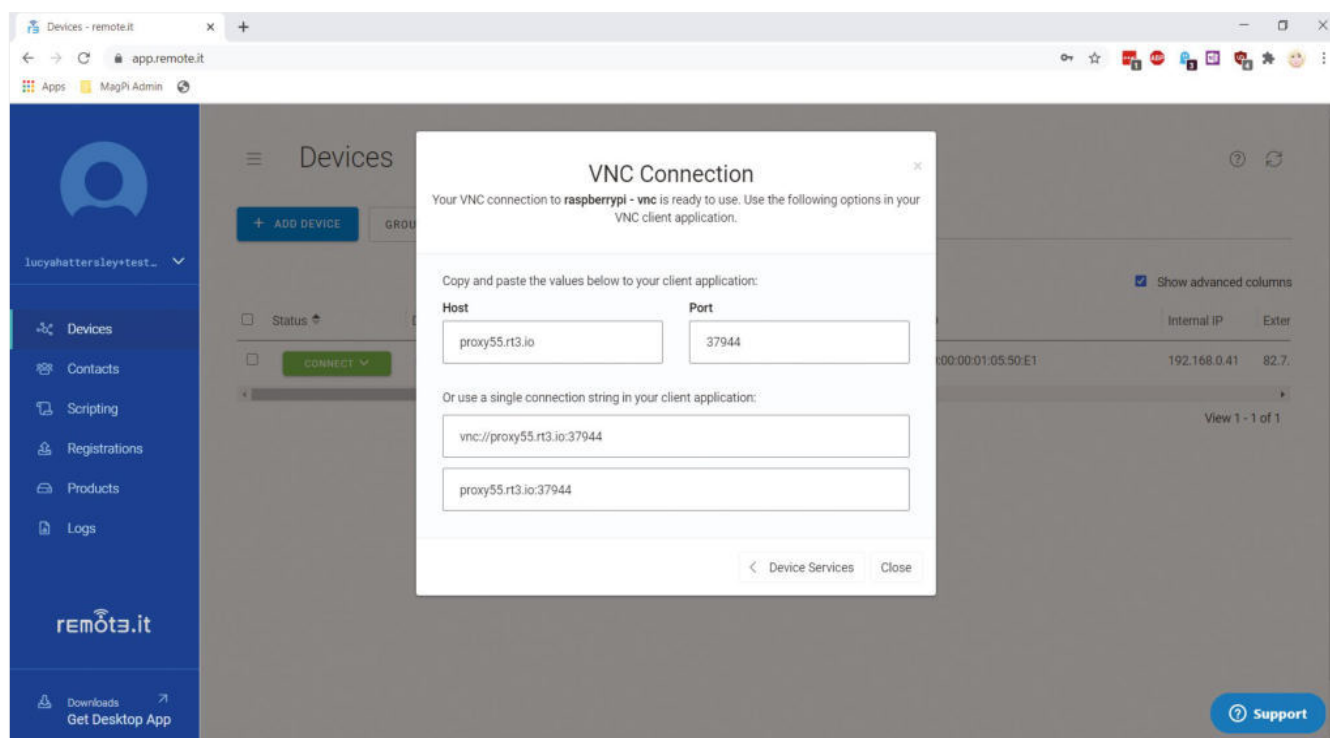
Double-click the icon and click Continue at the alert window. Enter 'pi' as the username and your Raspberry Pi OS password (from Step 2). Click OK to access the Raspberry Pi OS desktop remotely.

07 Connect via app

The drawback to the web service is that all your data passes through a proxy computer. For a more private connection, the remote.it app for Windows and macOS enables you to form a peer-to-peer (P2P) connection.

Data does not pass through remote.it servers (only the connection information), the data throughput is faster so the screen response time is better, and the connection will stay active as long as both devices remain connected. Finally, the URL remains the same and can be bookmarked (unlike the proxy connection which needs to be re-entered into VNC Viewer whenever you start the connection).

To create the P2P connection, you need to download remote.it's desktop software. Click on 'Get Desktop app' from the app.remote.it website (or visit remote.it/downloads) and click Download for Windows (or macOS, depending on your computer). Double-click the install file and follow the installation instructions. Open the remote.it software. You may see 'We need to install or update our service in order to maintain background connections.' Click on Install Service



▲ The remote.it website is used to open a proxy connection (with data routed through remote.it's servers)

and Yes to the User Account Control alert. Enter your remote.it account credentials into the app to sign in.

Under Devices, you will see your Raspberry Pi. Click it and click Connect next to VNC. The VNC icon will turn blue once connected.

You can launch VNC Viewer directly from the app (using the launch icon) or copy and paste the connection information into VNC Viewer (as in the previous step).

08 Cloaking your port

You can access Raspberry Pi remotely using remote.it. However, your VNC port is now open and listening. You can check this with the `lsof` (list of files) command:

```
sudo lsof -i -P -n | grep LISTEN
```

As well as the remote.it connection, it will show `vncserver` running on port 5900. The solution to this is to use a script, provided by remote.it, that cloaks this port. With the cloaking script active, people cannot scan your Raspberry Pi to find the open port because the port will not respond to

incoming traffic. Meanwhile, remote.it can get through because remote.it software establishes the connection.

“ You can use script remote.it with any language that runs on your Raspberry Pi ”

09 Upload scripts

You can use script remote.it with any language that runs on your Raspberry Pi. It's best to practise with a test script. Head to the Uploading a Device Script page on remote.it's website (magpi.cc/remotescripting) and click **show-device-info.sh** to download the file.

In the **app.remote.it** web interface, choose Scripting and click Upload. Change the option to 'Executable script or program' and click 'Choose file'. Use the File Explorer window to find the file and click Upload.

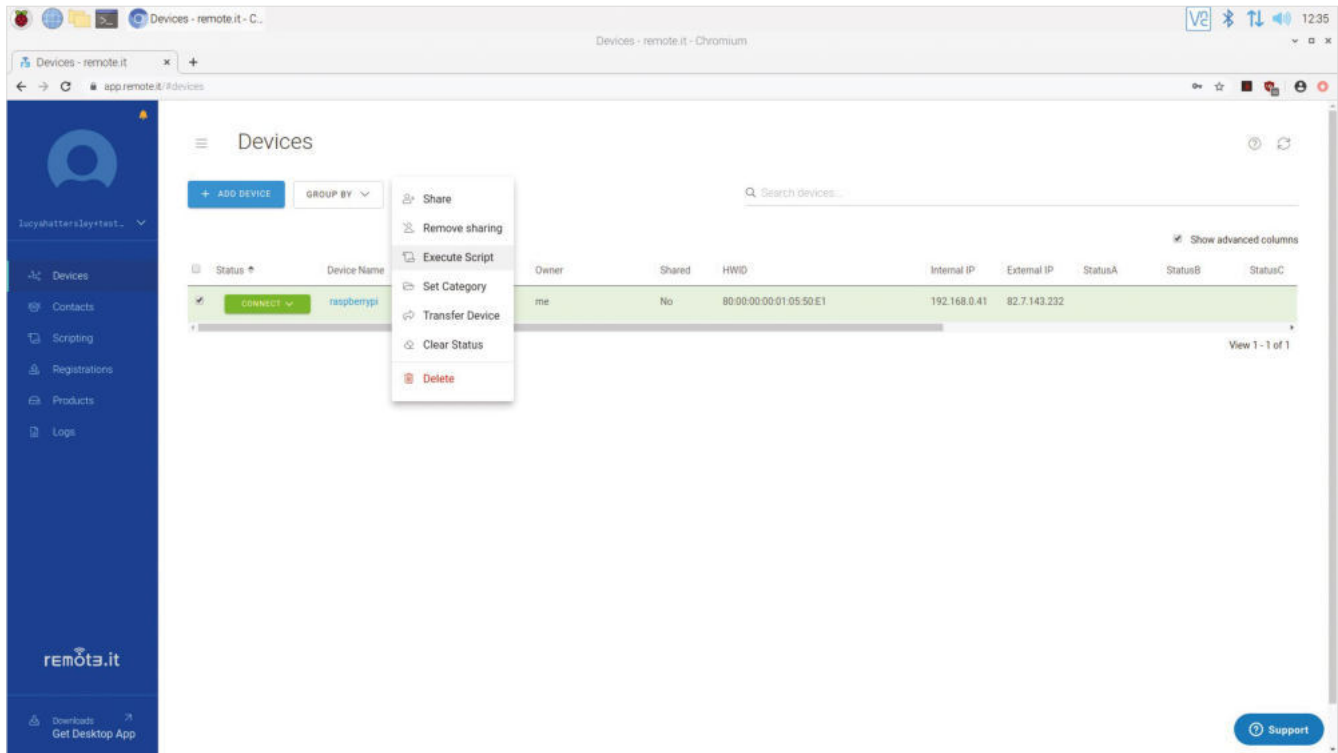
10 Running the script

Scripts are run from the Devices window. Place a tick in the checkbox next to your Raspberry

Top Tip

Create scripts

The remote.it service supports scripts in any language your Raspberry Pi system supports. The script is run just like any other executable on your machine. magpi.cc/remotescripting



▲ The cloaking script prevents VNC and SSH ports responding to requests (other than from remote.it)

Pi and put a check next to 'Show advanced columns'. (This will display the StatusA, StatusB, StatusC, and extra columns used by the test script). Click Actions > Execute Script; highlight **show-device-info.sh** and click Next.

Highlight the 'Check to allow all devices to update the status columns during Job execution' box. (This option displays information in the columns as the script runs.)

This time you'll get two options in the Bulk Execution window: 'Cloak Service' and 'Uncloak Service'. Highlight 'Cloak Service' and click Finish and Submit. Scripts can take some while to run. Choose Scripting in the sidebar and you will see the script progress. You can also cancel and delete scripts from this window.

On average it takes three minutes for the job status to update. So your script may actually complete and update the cells before the Job Status catches up. It should respond within three to four minutes.

Top Tip

Proxy vs peer-to-peer

Remote.it explains the difference between its proxy and peer-to-peer connections in this document: magpi.cc/remotep2p

“ The P2P service ensures that your data is not flowing through remote.it's service ”

11 The cloaking script

Now download the **cloak-vnc.sh** script from remote.it's GitHub page (magpi.cc/remotecloak). Upload it in the same manner as the **show-device-info.sh** script. And run it from the Devices window.

12 Cloaked and connected

Now connect to your Raspberry Pi via the remote.it app for a secure connection. Your port will be cloaked on Raspberry Pi and because you're connecting via remote.it's service, you will not be sharing your IP and port information publicly. And the P2P service ensures that your data is not flowing through remote.it's service. ”

CUSTOM PC

THE BEST-SELLING MAG FOR PC HARDWARE, OVERCLOCKING, GAMING & MODDING

THE MAGAZINE FOR

PC HARDWARE ENTHUSIASTS

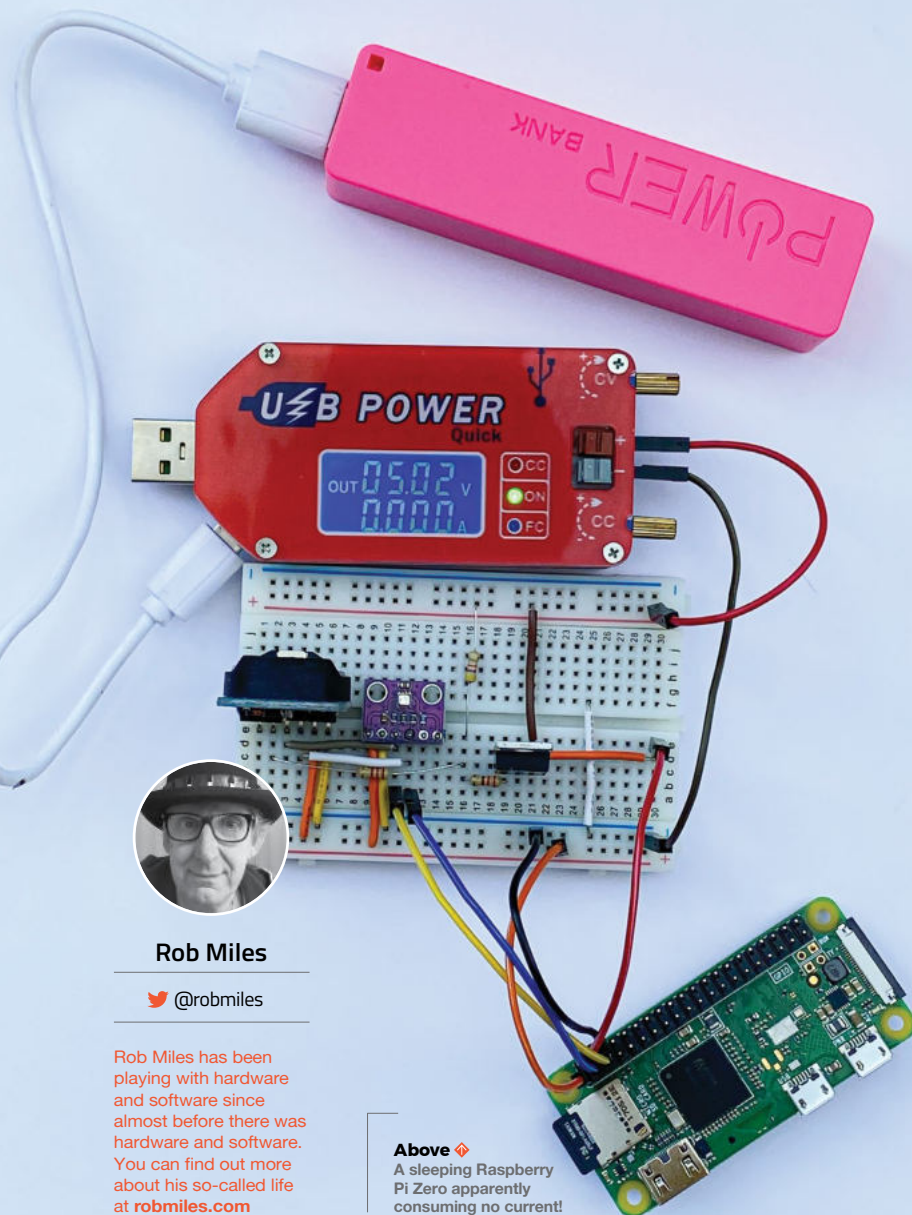


ISSUE 202 OUT NOW

VISIT **CUSTOMPC.CO.UK** TO LEARN MORE

Create low-power, clock-controlled devices

Consume close to zero power when your sensor isn't sensing



Rob Miles

🐦 @robmiles

Rob Miles has been playing with hardware and software since almost before there was hardware and software. You can find out more about his so-called life at robmiles.com

Above ♦
A sleeping Raspberry Pi Zero apparently consuming no current!

Do you want to make a sensor with a battery life you can measure in days rather than hours? Even if it contains a power-hungry device like a Raspberry Pi? By cunning use of a real-time clock module, you can

make something that wakes up, does its thing, and then goes back to sleep. While asleep, the sensor will sip a tiny amount of current, making it possible to remotely monitor the temperature of your prize marrow in the greenhouse for days on end from a single battery. Read on to find out how to do it.

REAL GOOD TIME CLOCK

We'll be using a DS3231 real-time clock which is sold in a module, complete with a battery. The DS3231 contains two alarms and can produce a trigger signal to control a power switch. To keep our software simple, we are going to implement an interval timer, but there is nothing to stop you developing software that turns on your hardware on particular days of the week or days in the month. The DS3231 is controlled using I²C, which means it can be used with lots of devices.

MOSFET ACCOMPLI

The power to our Raspberry Pi Zero is controlled via a P-channel MOSFET device operating as a switch. The 3.3V output from Raspberry Pi is used to power the DS3231 and our BME280 sensor. The gate on the MOSFET is connected via a resistor network to the SQW output from the DS3231.

You can think of a MOSFET as a kind of switch. It has a source pin (where we supply power), a drain pin (which is the output the MOSFET controls), and a gate pin. If we change the voltage on the gate pin, this will control whether the MOSFET conducts or not.



HackSpace

This tutorial is from in HackSpace Magazine. Each issue includes a huge variety of maker projects inside and outside of the sphere of Raspberry Pi, and also has amazing tutorials. Find out more at hsmag.cc

We use a P-channel MOSFET to switch the power because the gate voltage must be pulled down to cause the MOSFET to conduct, and that is how P-channel devices function.

MOSFET devices are all about voltage. Specifically, when the voltage difference between the source and the gate pin reaches a particular value, called the threshold voltage, the MOSFET will turn on. The threshold voltage is expressed as a negative value because the voltage on the gate must be lower than the voltage on the source. The MOSFET that we're using turns on at a threshold voltage of around -3.7 volts and off at a voltage of -1.75 volts.

The SQW signal from the DS3231 is controlled by a transistor which is acting as a switch connected to ground inside the DS3231. When the alarm is triggered, this transistor is turned on, connecting the SQW pin to ground. The diagram in **Figure 1** (overleaf) shows how this works.

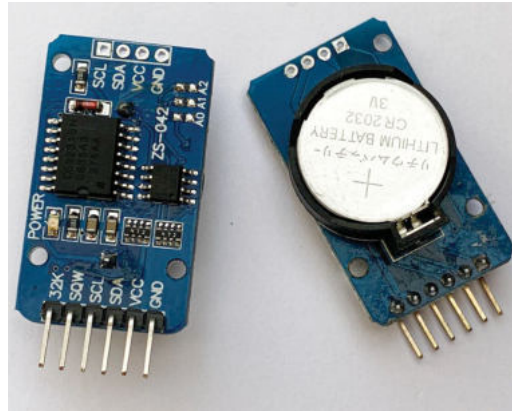
// The program does whatever it needs to do. This is the action that you want to perform at regular intervals **//**

The resistors R1 and R2 are linked to the supply voltage at one end and the SQW pin and the MOSFET gate on the other. When SQW is turned off the voltage on the MOSFET gate is pulled high by the resistors, so the MOSFET turns off. When SQW is turned on, it pulls the voltage on the MOSFET gate down, turning it on.

Unfortunately, current leaking through R1 and R2 to the DS3231 means that we are not going to get zero current consumption when the MOSFET is turned off, but it is much less than 1 milliamp.

POWER CONTROL

Now that we have our hardware built, we can get some code running to control the power. The DS3231 is connected to Raspberry Pi using I²C. Before you start, you must enable I²C on your Raspberry Pi using the `raspi-config` tool. Use `sudo raspi-config` and select Interfacing Options. Next, you need to make sure that you have all the I²C libraries installed by issuing this command at a Raspberry Pi console:



Above You can pick up one of these modules from lots of suppliers. Make sure that you get one with the SQW connection, as that provides the alarm signal

```
sudo apt-get install python3-smbus python3-dev  
i2c-tools
```

The sequence of operation of our sensor is as follows:

1. The program does whatever it needs to do. This is the action that you want to perform at regular intervals. That may be to read a sensor and send the data onto the network, or write it to a local SD card or USB memory key. It could be to read something and update an e-ink display. You can use your imagination here.
2. The program then sets an alarm in the DS3231 at a point in the future, when it wants the power to come back on.
3. Finally, the program acknowledges the alarm in the DS3231, causing the SQW alarm output to change state and turn off the power.

CLOCK SETTING

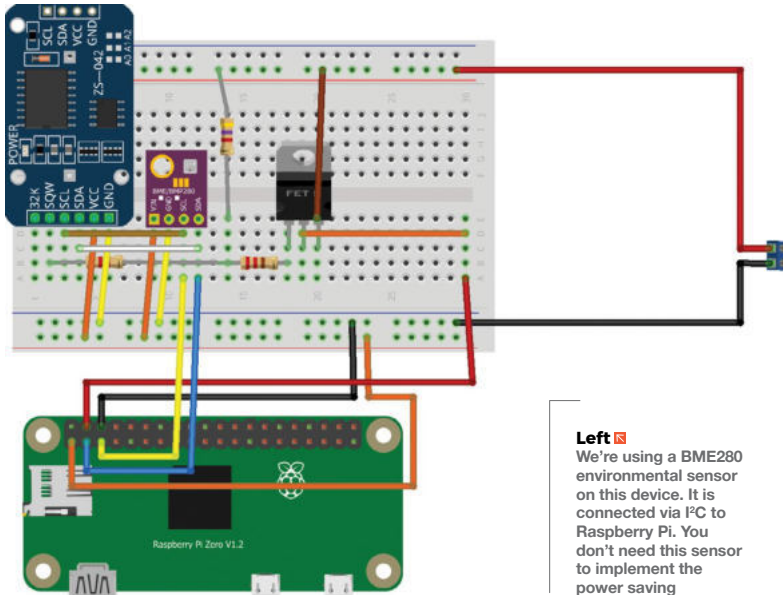
The program below only uses a fraction of the capabilities of the DS3231 device. It creates an →

DEBUGGING

It can be hard to debug your programs when your program keeps turning off the power to your device. To get around this, you can add a switch that connects SQW to ground. When you close the switch, it will turn on the MOSFET and power up your device.

YOU'LL NEED

- ◆ **DS3231 powered real-time clock module with battery backup**
Make sure it has a battery holder and an INT/SQW output pin
- ◆ **P-channel MOSFET**
The IRF9540N works well
- ◆ **Three resistors: 2.2 k Ω , 4.7 k Ω , and 220 Ω**
- ◆ **A device you want to control**
This can be a PIC, Arduino, ESP8266, ESP32, or Raspberry Pi. My software is written in Python and works in MicroPython or on Raspberry Pi, but you can find DS3231 driver software for lots of devices
- ◆ **Sensor you want to use**
We're using a BME280 to get air temperature, pressure, and humidity
- ◆ **Breadboard or prototype board to build up the circuit**



Left We're using a BME280 environmental sensor on this device. It is connected via I²C to Raspberry Pi. You don't need this sensor to implement the power saving

QUICK TIP

It is called the SQW pin because the DS3231 can be configured to produce a square wave signal on this pin which can be used by circuits needing a regular clock signal.

interval timer that can time hours, minutes, and seconds. Each time the program runs, the clock is set to zero, and the alarm is configured to trigger when the target time is reached.

Put the program into a file called **SensorAction.py** on your Raspberry Pi, and put the code that you want to run into the section indicated.

```
import smbus

bus=smbus.SMBus(1)

DS3231=0x68

SECONDS_REG=0x00
ALARM1_SECONDS_REG=0x07

CONTROL_REG=0x0E
STATUS_REG=0x0F

def int_to_bcd(x):
    return int(str(x)[-2:], 0x10)
```

MICROCONTROLLERS

You can also use CircuitPython or MicroPython to interact with the DS3231 over I²C. The I²C commands are slightly different, but the fundamentals are the same. Look in the GitHub repository for this article for more details.

POWER DOWN

The SensorAction program turns off your Raspberry Pi without shutting it down properly, which is something your mother probably told you never to do. The good news is that in extensive testing, we've not experienced any problems with this. However, if you want to make your Raspberry Pi totally safe in this situation, you should make its file system 'read-only', which means that it never changes during operation and therefore can't be damaged by untimely power cuts. There are some good instructions from Adafruit here: hsmag.cc/UPgJSZ.

Note: making the operating system file store read-only does not prevent you creating a data logging application, but you would have to log the data to an external USB key or SD card and then dismount the storage device before killing the power.

If you are using a different device, such as an ESP8266 or an Arduino, you don't need to worry about this as the software in them is inherently read-only.

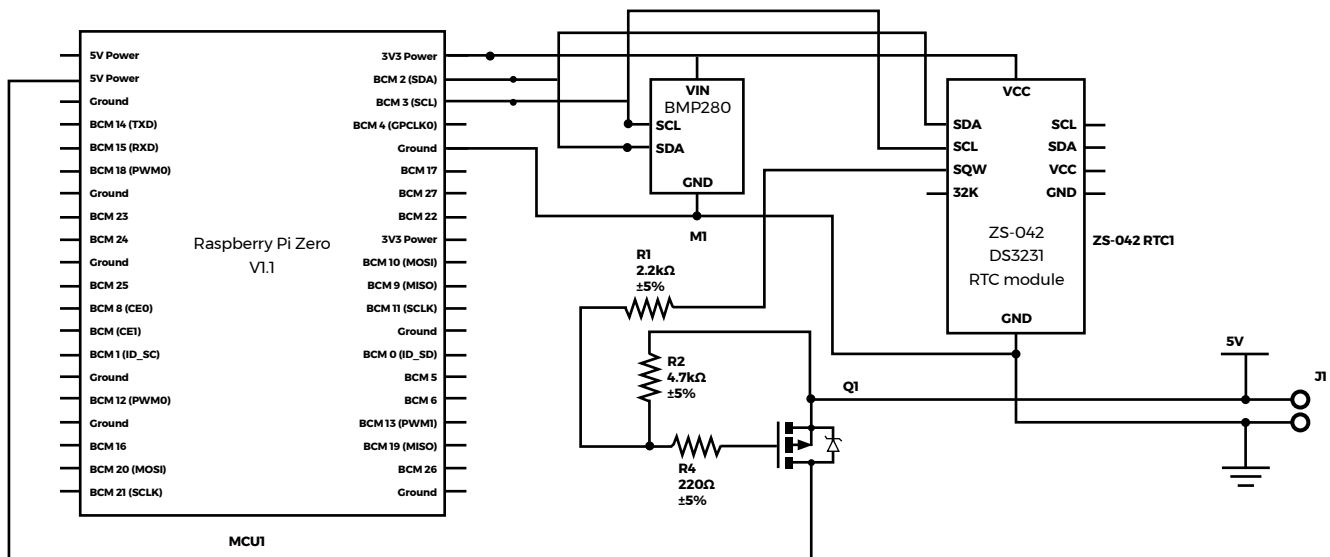
```
def write_time_to_clock(pos, hours, minutes, seconds):
    bus.write_byte_data(DS3231, pos, int_to_bcd(seconds))
    bus.write_byte_data(DS3231, pos+1, int_to_bcd(minutes))
    bus.write_byte_data(DS3231, pos+2, int_to_bcd(hours))

def set_alarm1_mask_bits(bits):
    pos=ALARM1_SECONDS_REG
    for bit in reversed(bits):
        reg = bus.read_byte_data(DS3231, pos)
        if bit:
            reg = reg|0x80
        else:
            reg = reg&0x7F
        bus.write_byte_data(DS3231, pos, reg)
        pos = pos+1

def enable_alarm1():
    reg = bus.read_byte_data(DS3231, CONTROL_REG)
    bus.write_byte_data(DS3231, CONTROL_REG, reg|0x05)

def clear_alarm1_flag():
    reg = bus.read_byte_data(DS3231, STATUS_REG)
    bus.write_byte_data(DS3231, STATUS_REG, reg&0xFE)

def check_alarm1_triggered():
    return bus.read_byte_data(DS3231, STATUS_
```



```
REG)&0x01 != 0
```

```
def set_timer(hours, minutes, seconds):
    # zero the clock
    write_time_to_clock(SECONDS_REG, 0, 0, 0)
    # set the alarm
    write_time_to_clock(ALARM1_SECONDS_REG,
        hours, minutes, seconds)
    # set the alarm to match hours minutes and
    seconds
    # need to set some flags
    set_alarm1_mask_bits((True, False, False,
        False))
    enable_alarm1()
    clear_alarm1_flag()
#
# Your sensor behaviour goes here
#
set_timer(1,30,0)
```

The `set_timer` function is called to set the timer and clear the alarm flag. This resets the alarm signal and powers off the sensor. The example above will cause the sensor to shut down for 1 hour 30 minutes.

ALWAYS RUNNING

To get the program to run when the Raspberry Pi boots, use the Nano editor to add a line at the end of the `rc.local` file that runs your program.

```
sudo nano /etc/rc.local
```

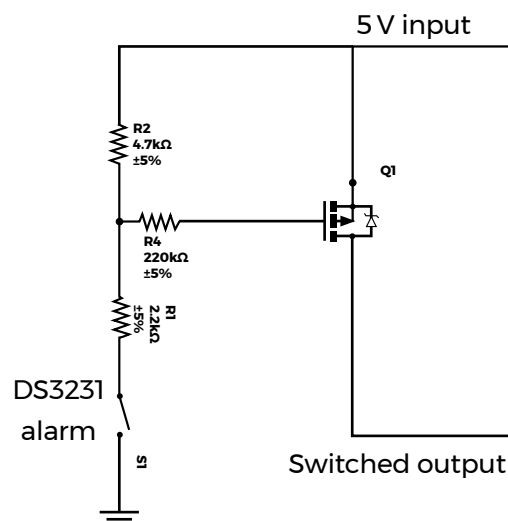
Use the line above at the command prompt to start editing the `rc.local` file and add the following line at the end of the file:

```
python3 /home/pi/SensorAction.py &
```

This statement runs Python 3, opens the **SensorAction.py** file, and runs it. Don't forget the ampersand (&) at the end of the command: this starts your program as a separate process, allowing the boot to complete. Now, when Raspberry Pi boots up, it will run your program and then shut itself down. You can find a full sample application on the GitHub pages for this project (hsmag.cc/Yx7q6t). It logs air temperature, pressure, and humidity to an MQTT endpoint at regular intervals. Now, go and start tracking that marrow temperature! ▣

Above ▣ You can use any other microcontroller that implements I²C

Figure 1 ▣ The SQW output from the DS3231 will pull the gate of the MOSFET low to turn on the power to Raspberry Pi



QUICK TIP

Don't try to use an N-channel MOSFET. They switch on when their input gate is pulled high, which is no good to us.

Supercharged Scope



Mike Cook

Veteran magazine author from the old days, writer of the Body Build series, plus co-author of *Raspberry Pi for Dummies*, *Raspberry Pi Projects*, and *Raspberry Pi Projects for Dummies*.

magpi.cc/mikecook

This new Arduino / Raspberry Pi oscilloscope will sample signals at 1MHz, great for audio work and even supersonic signals

01 Supercharging

Back in *The MagPi* #71 (magpi.cc/71), we showed you how to make an oscilloscope using an Arduino Nano, to handle sample acquisition, and a Raspberry Pi to display and analyse the results. This worked well and allowed many people to look at waveforms for the first time without needing expensive equipment. What was pleasing was the rapid real-time response of the display. The only limitation was that the sample rate was capped to the speed of the Arduino's internal A/D converter at 56 ksps (kilosamples per second). So now we are supercharging this project, and getting sample rates of up to 1Mps.

02 How do we do this?

We achieve this supercharging by using an external, fast, A/D (analogue to digital) converter, the AD7822. It uses an 8-bit half flash converter to achieve a conversion time of 420 ns or just over 2Mps. A flash converter is a circuit that uses a bunch of voltage comparators to find out what an unknown voltage is. We used a flash converter back in 2016 in *The MagPi* #48

▼ **Figure 1** A 2-bit flash converter, sometimes called a continuous converter

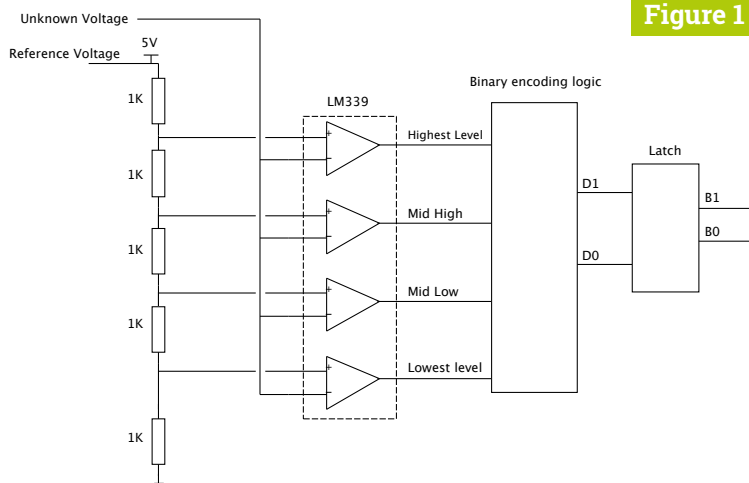


Figure 1

Swimming Simulator project (magpi.cc/48).

Figure 1 shows you the basic idea for splitting out an unknown voltage into four levels, or two bits of resolution. This is fast because you only need to wait for the settling time of the comparators to get your reading.

03 Why are flash converters rare?

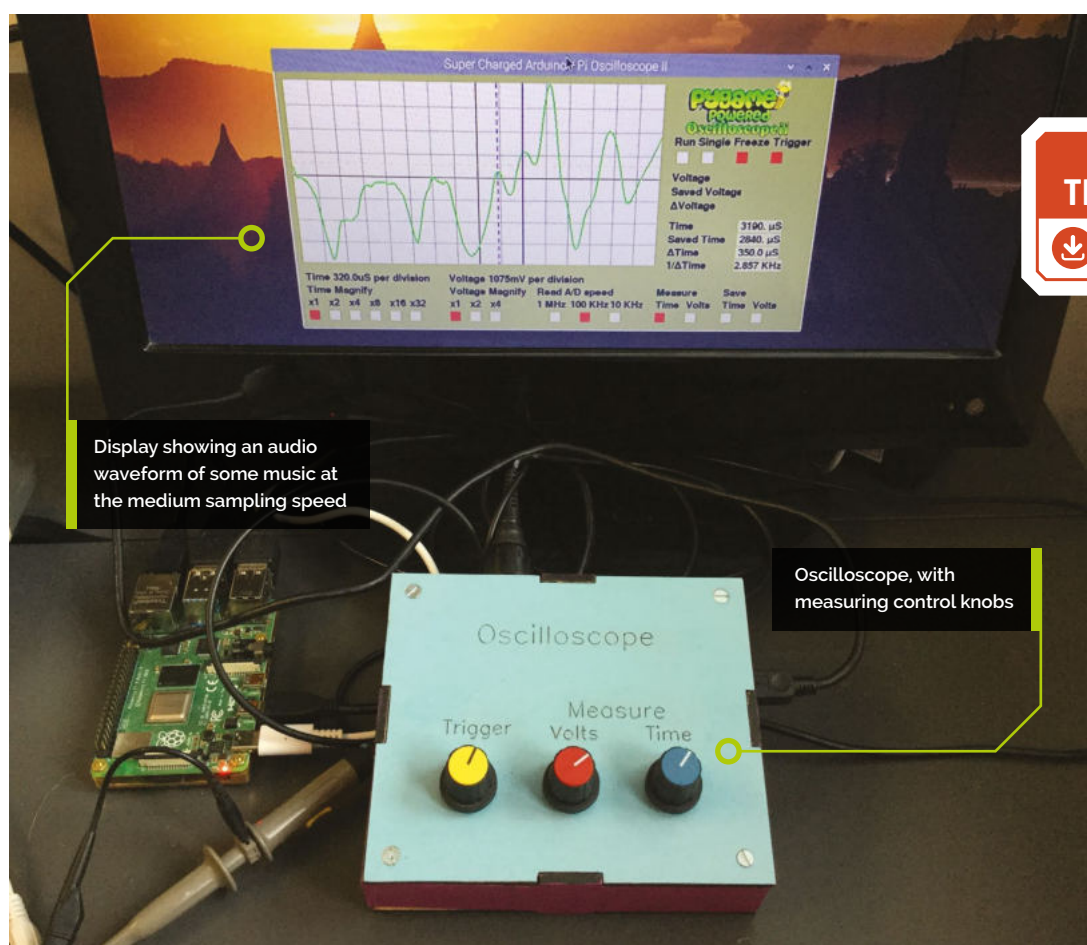
We don't use flash converters very often, because they are expensive to make. You see, the number of comparators you need rises exponentially with the A/D resolution. Each extra bit of resolution requires a doubling of the number of comparators you need. So to get 8-bit resolution you need 256 comparators, which is what is used in video digitisers. The AD7822 employs a half flash technique, which means it uses a 4-bit flash converter, which only needs 16 comparators, and internally switches a reference voltage to make two flash measurements and combines them into a single 8-bit reading.

04 Using the AD7822

The AD7822 has a parallel output. That means it requires a separate input on the Arduino for each bit of the reading. To start the conversion process, we must give a pulse on the ~CONVST (conversion start) pin of the chip. Note that placing '~' in front of a pin name means that the action is true when the pin is a logic zero. When the conversion is finished, the chip puts a zero on the ~EOC (end of conversion) pin. There are other pins for power down and output enabling, but we don't need to consider these.

05 AD7822 analogue span

The analogue input of the AD7822 has a single analogue input; other variations of this



Display showing an audio waveform of some music at the medium sampling speed

Oscilloscope, with measuring control knobs

**DOWNLOAD
THE FULL CODE:**



magpi.cc/pibakery

You'll Need

- AD7822 A/D converter
magpi.cc/ad7822
- MCP602-I/P operational amplifier
magpi.cc/mcp602
- Arduino Nano board
magpi.cc/arduino nano

chip have four or eight inputs. The input span – the input voltage range it converts into valid readings – is 2.5V, but this span can be anywhere between the supply voltage rails. Exactly where it is can be controlled through the Vmid pin, if it is used as an input. This pin can also be used as an output, and then it will output a voltage that is halfway between ground and the voltage reference. In our design, we use this pin to drive the input operational amplifier.

06 What limits the sample rate?

The sample rate is limited not so much by the speed of conversion, like it was in our original oscilloscope, but by how long it takes the Nano to read in the data and store it somewhere. This is a bit longer than it takes to make the conversion, hence we can only get a sample rate of about 1 Msps despite the conversion being done at 2.38 Msps. We need to use direct port addressing on the Nano to get this speed, but due to the way the Nano is designed we have to combine the readings from two ports, shown in **Figure 2**.

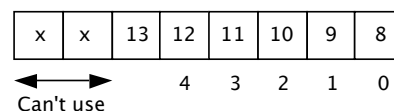
07 The data acquisition system

The schematic for the oscilloscope is shown in **Figure 3** (overleaf). It differs from our original oscilloscope in two ways. First, most obviously, is the addition of the AD7822 fast A/D chip; second is the use of an operational amplifier on the front end, allowing us to have a higher input impedance than the original oscilloscope, and a DC coupled input. This means that the display will show any

▼ **Figure 2** Direct port addressing pin assignments for this project

Figure 2

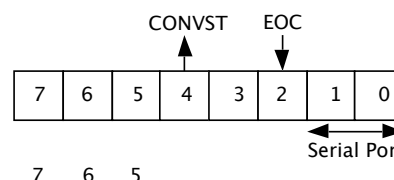
Port B



Data Pin Number

A/D Data bit

Port D



Data Pin Number

A/D Data bit

A/D read = Port B & 0x3F | Port D & 0xC0

MAGDOWNLOAD.ORG

FREE DOWNLOAD

LATEST MAGAZINES

HIGH QUALITY TRUE-PDF

MAGDOWNLOAD.ORG

Figure 3

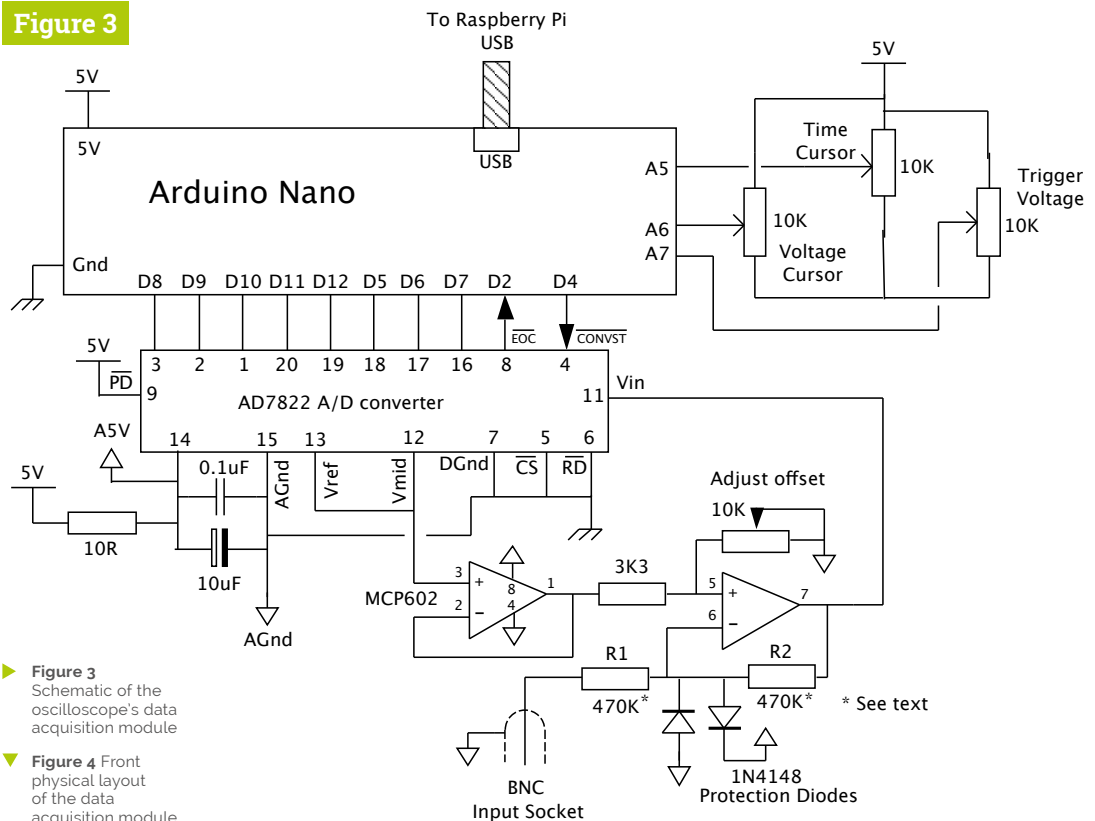


Figure 3
Schematic of the
oscilloscope's data
acquisition module

Figure 4
Front
physical layout
of the data
acquisition module

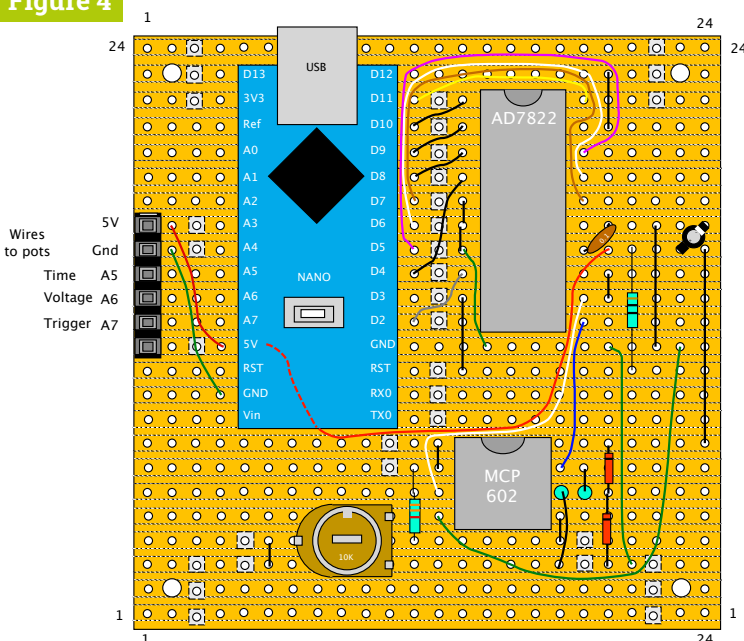
Top Tip

Do not connect to the mains

This oscilloscope, along with many other commercial oscilloscopes, must not be used to make direct measurements of the mains electricity supply. See this video for a full explanation: magpi.cc/oscsafety.

“ The characteristics of the MCP602 define the input and frequency response of the entire oscilloscope ”

Figure 4



DC component of the signal we are measuring, as a shift up or down. The MCP602 operational amplifier is used to buffer the input and adjust the signal's DC offset.

08 MCP602

The characteristics of the MCP602 define the input and frequency response of the entire oscilloscope. While this is a good op-amp, all op-amps are a compromise between performance and price. It has good rail to rail output capabilities for our design but you could ask for a better bandwidth product (GBW). This GBW defines how the open loop gain of an amplifier changes with frequency, so that if you have a GBW of 2.8MHz, like the MCP602, then the gain drops off to one when the input signal is 2.8MHz. The MCP6022-E/P has a 10MHz bandwidth, but we have not tested this yet.

09 Construction

We built the circuit on a 24 rows by 24 holes piece of stripboard. **Figure 4** shows the layout. The breaks on the underside tracks are shown in **Figure 5**. We soldered header pins into the Nano and mounted it on socket strips soldered into the

stripboard. A six-row socket was used to connect this board to the three control potentiometers mounted on the lid of a box (**Figure 6**). We made a laser-cut box from 3.2mm MDF, but you can use any pre-made box – a metal one would be better if you have the choice. **Figure 7** shows it being mounted in the box.

10 The software

The software for this project is on GitHub (magpi.cc/pibakery). Three extra buttons added to the original oscilloscope display screen set the sample rate at 1MHz, 100kHz, and 10kHz. The faster rate is so fast that we have to change the trigger strategy. You can't just monitor the input until the trigger transition is reached and then gather the data, because too many samples will be missed. So instead of this we read in a double-size buffer's worth of data and search the first half for a transition. When we find it, or not, we send a buffer's worth of data from that point.

11 Performance

While the sample rate is 1Mps, that doesn't mean you can display a 1MHz signal: the top frequency display depends on how many samples you need on a waveform to represent its shape. At 20 samples per cycle the waveform looks good, so a 50kHz waveform can be displayed no problem. If you care to push this with time expansion then you can measure frequencies of 200kHz, but the waveform will start to distort due to samples being picked up asynchronously to the wave you are measuring. Pressing **S** on the keyboard grabs a screenshot of the scope.

12 Improvements


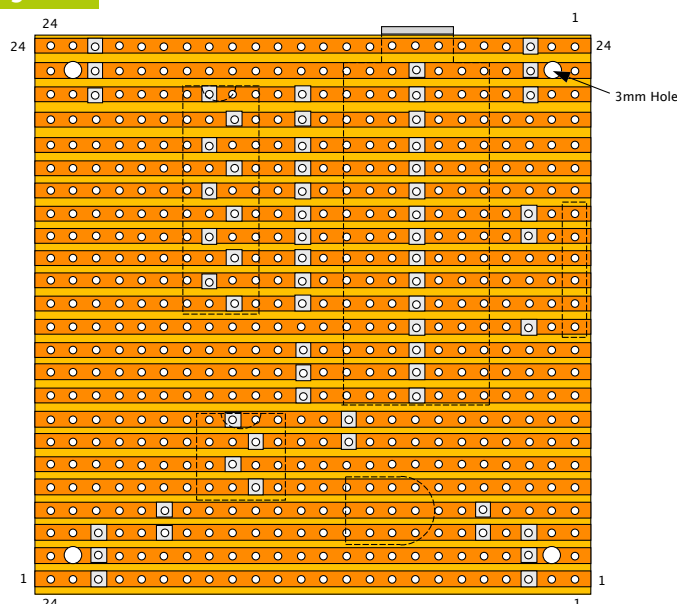
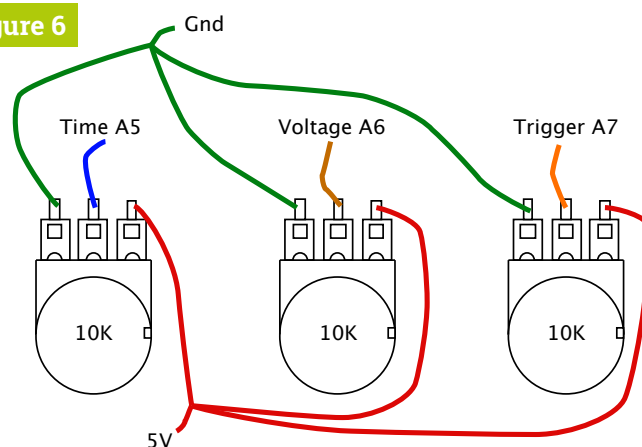
You can make improvements to this design in several ways. First, you might want to add an AC coupling option. This is simply a capacitor of about 10nF or smaller in series with the input, with a switch option to short it out. You can add a shift input pot to add an offset to the waveform. You could add a multi-position switch to change the values of R1 and R2 to alter the gain and get a bigger or smaller voltage range, and use a spare analogue input on the Nano to signal the gain back to the software. 

Figure 5



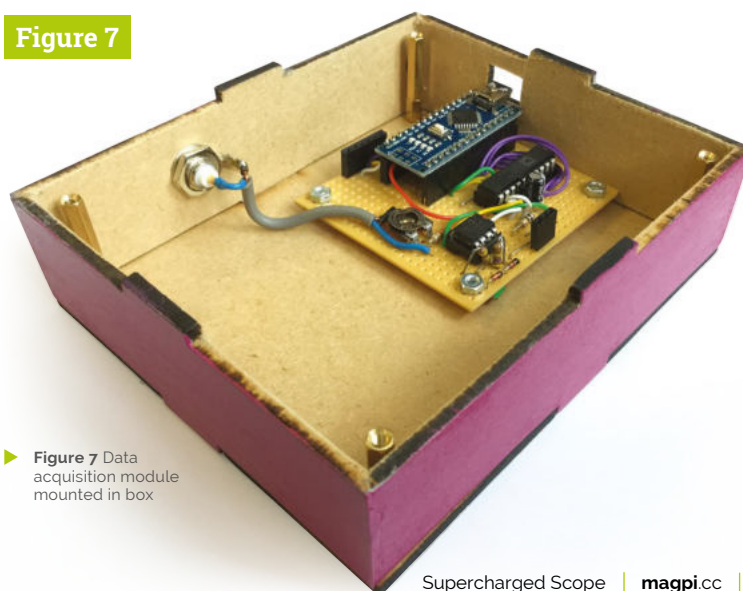
► **Figure 5** Back track breaks on the data acquisition module

Figure 6



► **Figure 6** Wiring the oscilloscope's control knobs

Figure 7



► **Figure 7** Data acquisition module mounted in box

Make making accessible

How you can make sure everyone is able to join the fun

Everyone should have a chance to learn about programming and digital making, whatever their circumstances. We live in a time where access to free resources on almost every aspect of computing, programming, and making is widespread. Just because they're free, though, does not mean everyone can use them.

Not everyone can use the internet or physical spaces in the same way. This is where the concept of accessibility comes in: allowing more people to freely access your content.

Over the next few pages, we're going to give tips on how to make sure resources can reach the widest possible audience.

Accessibility on computers

Things to consider when creating digital content

Web and UI

Fonts

Font choice for a resource is very important – you want it to be readable after all. A sans serif font (i.e. one without extra strokes, such as Helvetica or Comic Sans) is much preferred. Don't just stop there, though. While Comic Sans itself is much derided in the design community, it's widely regarded as a more accessible font for those with dyslexia. There are other fonts with similar traits, such as Myriad Pro. Font size is also important. 12 to 14 point is a recommended minimum size, and most browsers allow for zooming in now as well.

FURTHER READING:

British Dyslexia Association – magpi.cc/bda

TIP!

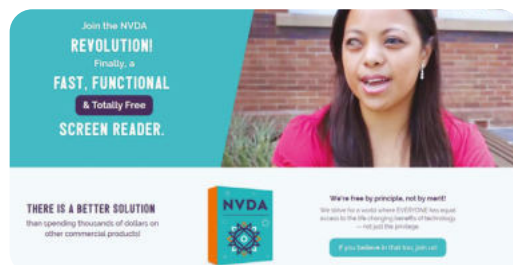
Anything with audio should have subtitles or be transcribed – this helps those with hearing impairments and others who might just be in a cafe. YouTube can help with subtitles, but you will need to fine-tune them.

SCREEN READER-COMPATIBLE

Blind or partially sighted folks use screen readers to actually read your website. As long as you're using text in a resource, the screen readers will be able to pick them up and read them. However, there are other things to consider. Don't put important text in images that are required for a resource. Any non-decorative pictures you do use should be fully described using the 'alt' tag on images. Also, don't use special characters if possible, and try to avoid emojis.

FURTHER READING:

WebAIM – magpi.cc/screenreader



SCREEN CONTRAST

Articles and resources on computers usually come in two flavours: black text on a white background, or white text on a black background. These are readable as they have very high contrast between the shades. However, using pure contrast like this can cause eye strain for many. Conversely, too low a contrast makes it unreadable for others.

If you plan to display text on a background, it's a great idea to check the contrast ratio using a tool such as the WebAIM contrast checker:

magpi.cc/contrastchecker.

FURTHER READING:

WebAIM – magpi.cc/contrast

CONSISTENCY

Good design is generally consistent, whatever the topic covered. Consistency in resources also helps a lot for accessibility – consistent use of language and design improves readability for those with Autistic Spectrum Disorder (ASD).

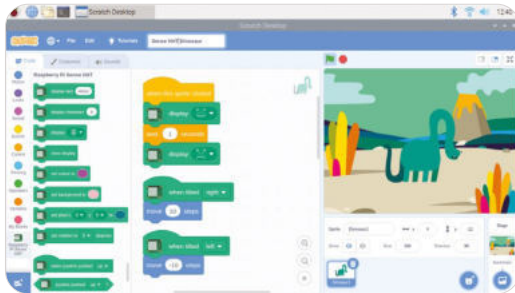
KEYBOARD NAVIGATION

Some people can only navigate via a keyboard or other similar input device. Using focus indicators in a logical reading order allows people to key through a page.

FURTHER READING:

WebAIM – magpi.cc/keyboardnav

Community and software



BLOCK PROGRAMMING FOR NON-ENGLISH SPEAKERS

Block-based visual programming languages like Scratch are great for teaching people the basics of programming. They're also very useful for students who are learning English as a second language. This way they don't have to worry about exact spellings of specific words, functions, and variables, and can still put code together.

FURTHER READING:

Block-based coding – magpi.cc/blockbased

MODERATE ONLINE SPACES

When we create an online space for a community, we have a responsibility to keep it safe and open to everyone. Moderation and published guidelines are key, and you shouldn't be afraid to remove people from the community if they are causing trouble or making people uncomfortable.

FURTHER READING:

Creating a Safe Digital Space – magpi.cc/safedigitalspace

LIMITING NEW CONCEPTS

In education there is a term called 'cognitive load', which refers to the amount of new data and concepts that students need to process during a lesson. If there are too many concepts, students can experience cognitive overload which hampers learning. Keep this in mind when breaking any longer pieces up into chapters.

FURTHER READING:

Hello World 11 – magpi.cc/helloworld11

TIP!

Write as simply and as neutrally as possible. This allows new learners to understand without having to translate jargon, and can help people whose first language is not English.

Essential reading

While we've highlighted some important things here, it doesn't cover everything. Here are more resources to read up on.

Web contents accessibility guidelines (WCAG)

Created by many people and organisations, the goal of WCAG is to create a standard of accessibility for webpages.

magpi.cc/wcag

The A11Y Project

Resources for web developers on how to implement accessibility and check for it.

a11yproject.com

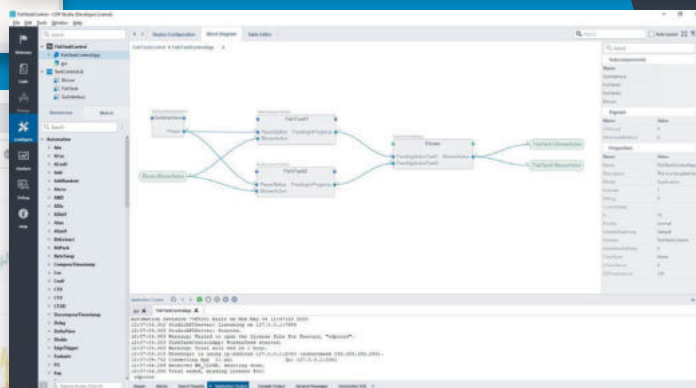
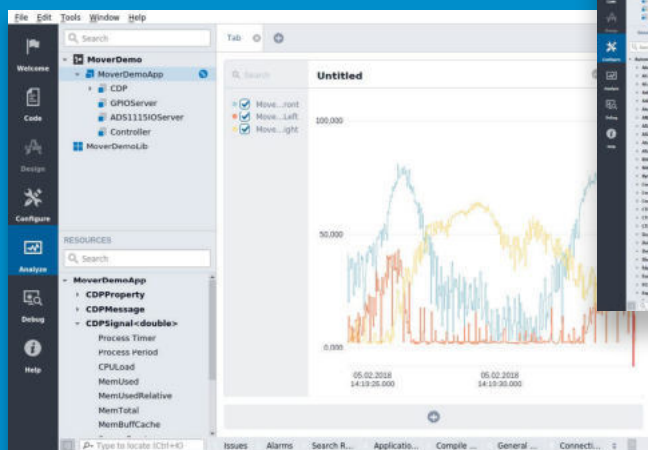
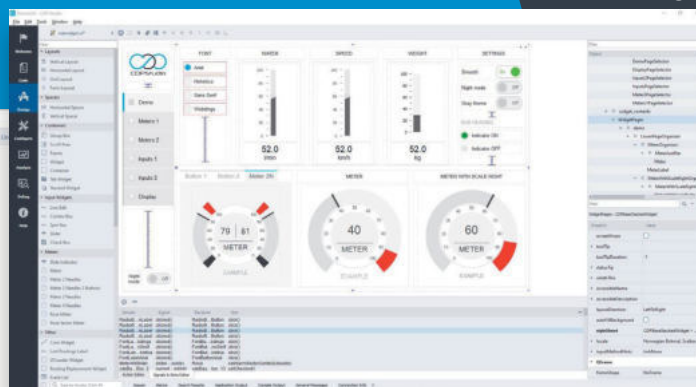
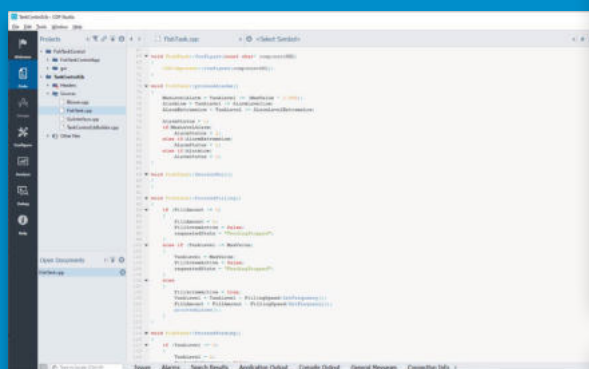
Tools of a blind programmer

Written by Parham Doustadar, this is a first-hand account of how he works as a blind programmer.

magpi.cc/blindprogrammer

Design

Code



Configure

Analyze

CDP STUDIO: HOME PROJECTS MADE EASY

Professional control system development tool

CDP Studio is a great software development tool for your home projects. Build systems for Raspberry Pi, use C++, open source libraries, out of the box support for GPIO, I2C, MQTT, OPC UA and more. Create user interfaces in the integrated designer. Code less and do more!

NOW FREE FOR NON-COMMERCIAL USE.

CDP Technologies AS
Hundsværgata 8, 6008 Ålesund, Norway
Tel: +47 990 80 900 • info@cdptech.com
www.cdpstudio.com



Accessibility in the real world

Creating an accessible and welcoming physical space

Preparing a space



WHEELCHAIR USERS AND OTHER MOBILITY IMPAIRMENTS

A lot of newer buildings, and pre-existing buildings, have ramps and lifts in them which allow everyone to move around them. Keeping spaces under desks and walkways clear is good as well.

Choosing a location because of wheelchair users (or those with other mobility impairments) is not always feasible due to cost and availability. However, you can get portable ramps that will work in some situations.

FURTHER READING:

UK accessibility regulations – magpi.cc/accessibilityregs

PUBLISHED, REGULAR SCHEDULES

People with ASD can sometimes be uncomfortable with an unfamiliar setting and unfamiliar schedule. Publishing a schedule online to an event page helps people prepare. Keeping this schedule for subsequent events allows a routine to follow, which can increase comfort.

More essential reading

CoderDojo Accessibility Guide

Official guide from the folks at CoderDojo on how you can make a space more accessible.

magpi.cc/cdaccessibility

ReCreate

Reusing and recycling materials for teaching, and their benefits for some SEND students.

magpi.cc/recreate

TIP!

KIBO is a robot that can be programmed using physical cards that are scanned by the robot: magpi.cc/kibo

TIP!

Allow for more time for people to complete the task than you think.

Teaching in the space



MENTORS

Some children, especially if they're classed under special educational needs and disability (SEND), can benefit greatly from a mentor or another form of one-to-one teaching and supervision.

A mentor can help guide folks in the right direction. Parents and guardians may also want to get involved, allowing them to aid in supervision.

FURTHER READING:

Hello World 11 – magpi.cc/helloworld11

PHYSICAL ALLEGORIES

When teaching programming concepts, having something outside of the computer can help those that don't speak English natively or have visual impairments. Code Jumper is a series of physical blocks that can be put together to create code, akin to a real-life version of Scratch.

For ESL folks, and even students in general, teaching examples using lines of people and using familiar contact and sensory approaches can help reinforce learning.

FURTHER READING:

Code Jumper – codejumper.com

TIP!

When giving people activities, don't be afraid to have cheat sheets available with helpful info.



PARTNERS FOR ESL

Another suggestion for people who are non-native English speakers is to pair them up with someone who is. Paul Gordon (@TeachtheTech) suggests a 'co-pilot' system, with a driver and navigator approach. This not only helps with computing, but can also help improve English capabilities.

Try not to pair up people who don't like each other, though.

TIP!

Remember, one approach for one person will not work for everyone.

PiCar-V Kit V2.0

SPECS

ELECTRONICS:

Robot HAT with PCF8591 ADC chip, PCA9685 PWM driver, TRAg118A motor driver

POWER:

18650 battery holder

MOTORS:

SF006C servo motors, F130SA-11200-38V DC gear motors

► SunFounder ► magpi.cc/picarv ► £100 / \$115

This detailed robot car build will put your assembly skills to the test. By **Lucy Hattersley**

We love a good robot here at *The MagPi* magazine, and we've tested out dozens of builds. With its smart looks, front-steering mechanism, and camera mount, SunFounder's PiCar-V grabbed our attention. We just had to review it.

It comes in at a reasonable price, too. It seems that the £100/\$115 mark is a good entry point for a bring-your-own Raspberry Pi robot kit. This build offers a lot of components for the money.

Many robots use the two wheels of a DC motor to provide tank-like steering. PiCar-V 2.0 takes things up a gear with an additional servo motor to steer the front wheels via a rack mechanism (called the Servo Linkage Plate).

A further two servos provide a pan-and-tilt mechanism for the included USB webcam (not an official Raspberry Pi Camera Module, unfortunately).

You have to bring your own Raspberry Pi board and microSD card to the table, plus two 18650 rechargeable Li-ion batteries (3.7V) and a charger. Even so, there's a lot here for your money.

Days to build

Oh, my days, there are so many items in the box. By our count, there are 128 separate components, including four wheels, three servos, two DC motors, the chassis, battery back, three HAT boards, and a wild amount of screws, standoffs, and nuts.

The very first step for each front wheel involves the wheel, a flange bearing, three connector plates, three more hex fixing plates, an M4×25 screw, and an M4 self-locking nut.

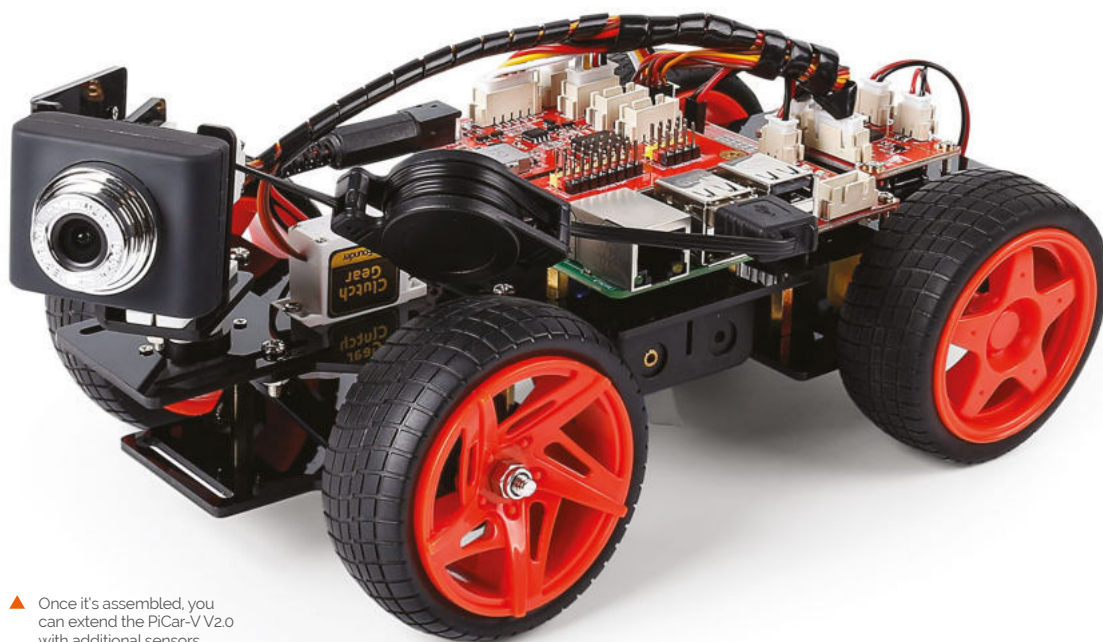
Everything you need is in the box, including a screwdriver, socket wrench, and two small wrenches. Fortunately, there's an 88-page full-colour booklet to walk you through the build and usage.

Even so. This is a taxing build. The assembly guide is available online as a PDF (magpi.cc/picarmanpdf), so why not take a good look through it to gauge your comfort factor with the build process before buying?

► The kit contains a huge array of components to assemble



▲ The final robot is good-looking, with a range of useful features to experiment with



▲ Once it's assembled, you can extend the PiCar-V V2.0 with additional sensors

We found the build straightforward, if fiddly. The two exceptions being attaching the front wheels to the steering mechanism, and screwing on the plates to the rotating servo mechanism (without rotating the servos themselves).

It's mostly time-consuming. Like jigsaws, Lego, and Meccano kits, there's a lot of fun to be had in long and detailed builds.

The setup regarding Raspberry Pi OS is a little outdated (recommending NOOBS or Balena Etcher, while Raspberry Pi now recommends the purpose-built Raspberry Pi Imager (magpi.cc/imager) tool. But the steps still work, and it walks you through setting up Raspberry Pi OS, enabling SSH (magpi.cc/ssh), and connecting to the PiCar-V robot over your local network.

Bring it together

The result is a good-looking car, if imperfect. The design prevents access to the HDMI port once PiCar-V V2.0 is assembled, so you can't attach a screen to debug Raspberry Pi OS if you lose the SSH connection. The design also means that the physical build is interrupted halfway as you set up the software on Raspberry Pi.


Once complete, the frame is bendy and, in truth, we'd prefer the Upper Plate to have a little more heft. We didn't encounter any problems in use, however.

“ There's a lot of fun to be had in a long and detailed build ”

There's also a nest of wires between the boards, and the USB camera cable extends over the front-left wheel. Some cable ties will come in handy.

Software support is extensive and everything is available on the GitHub repo (magpi.cc/picargit). You'll need to install Python and use pip to install PyQt5 and the requests library on Raspberry Pi.

When all is set up, you can run the server on Raspberry Pi and run the client software on your PC. The client software displays the webcam view, and you control the rover with **W, A, S, D** and the camera pan and tilt with the arrow keys. Android and iOS apps are available.

There is an example Python program that uses OpenCV to track a red ball. SunFounder has also provided a tutorial for programming the robot with DragIt (a Snap-based visual programming tool). You can read this, and lots of other documentation on the SunFounder Learn site (magpi.cc/picarlearn). When everything is finished and you're done with the example code, there are eight digital and four analogue channels on the Robot HAT, plus two spare I²C ports. Plenty of space for expansion with extra sensors. 

Verdict

We like this kit, but it has its flaws. You get a lot of components and some pieces of kit should be of a higher quality. It requires patience, but the supporting documentation is excellent. A nice robot car with potential for expansion.

8/10

UCS Universal Housing

SPECS

MATERIALS:

Polycarbonate
PC, polyamide
PA (0.3 to 3 mm)

PROTECTION:

IP40

FLAMMABILITY:

UL 94 (reduced
V2 when
stacked)

► Phoenix Contact ► magpi.cc/universal housings ► From £35 POA

If you're looking to deploy Raspberry Pi in an industrial or commercial setting, Phoenix Contact has you covered. By **Lucy Hattersley**

Increasingly, Raspberry Pi computers are turning up in industrial settings, from factories to warehouses, transport hubs, and retail environments.

Phoenix Contact is one of the biggest names developing a casing solution geared towards an industrial user. Its UCS (Universal Case System) Universal Housings are designed to contain Raspberry Pi in an industrial environment, and come in a range of sizes with flexible mounting and various configurations.

It comes with an IP40 rating: "Protected from tools and small wires greater than 1 millimetre. Not protected from liquids."

The modular cases come with either a solid lid or a side panel for an embedded display.

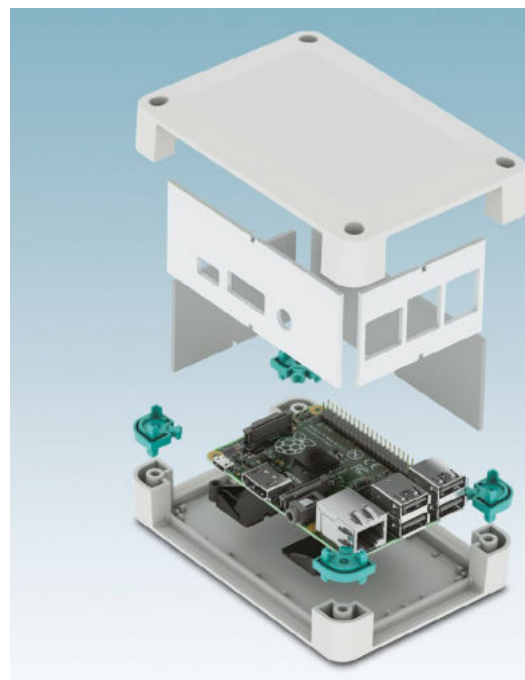
We got three sizes to experiment with: 125×87 mm, 145×125 mm, and 237×195 mm

"Universal Housings are designed to house Raspberry Pi in an industrial environment **"**

Made to measure

Matching accessories (which we haven't tested) include stands, wall mounts, stacking adapters, and fibre-optic light guides – much of what you need to deploy multiple Raspberry Pi computers in a range of environments.

Inside the cardboard box, we received two half-shells, four side panels (with openings for the Raspberry Pi connectors), adhesive pads to stick the Raspberry Pi to the shell, and Torx screws.



▲ The Universal Housing encases Raspberry Pi inside two shells and four panels

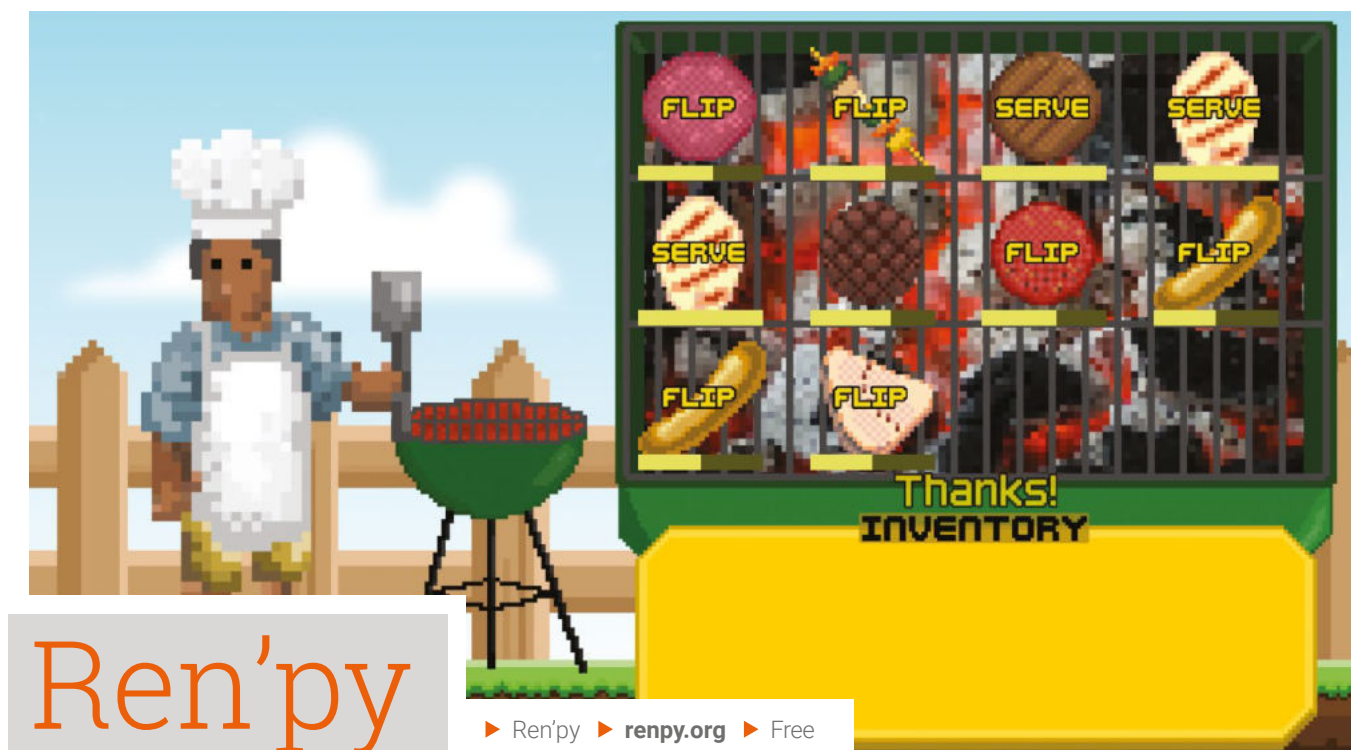
We used a Torx T7 screwdriver to fix the adhesive pads to our Raspberry Pi 4 and a Torx T10 to screw the two outer shells together. The presence of Torx screws will help prevent prying hands from getting into the shell, and the microSD card is tucked safely inside.

Documentation is excellent, and a data sheet (magpi.cc/ucsdnloads) covers technical data and assembly instructions. On the whole, we like the build process and the case has an aesthetically pleasing style that lifts it above its industrial roots. **"**

Verdict

Build quality is excellent and the versatility of applications will ensure the UCS Universal Housing finds itself happy in a wide range of industrial settings.

8/10



Ren'py

► Ren'py ► renpy.org ► Free

A visual novel game engine that's created in Python. **Rob Zwetsloot** tries some game dev

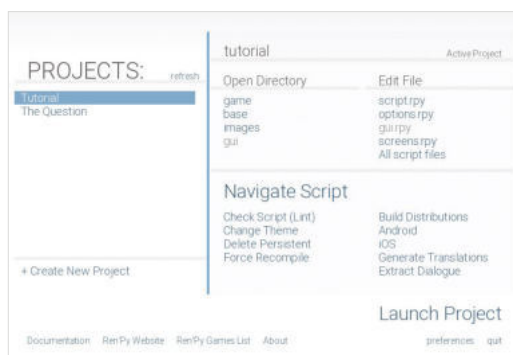
Verdict

A great starting point to learn about game development, and even get your first game out into the world.

9/10

Creating video games in 2020 may be easier than it was 20, even ten, years ago but it's still hard. As with a lot of skills, sometimes you need to practise with something similar and work your way up. Created a game in Scratch? Great. Printed Hello World in the Python shell? Wonderful. What's the next step, though?

We think Ren'py might be a good answer. Not because it's mainly used for creating visual novels (VN) – while easy to deride, they're a legitimate genre and difficult to make in their own way – but because it's designed to make it easier to develop. Not only does it have great starter guides, but it also has very advanced functions for more complex game elements, like game-generated animations.



- ▲ We created a cooking game, BBQ DAD, with Ren'py that works just fine on Raspberry Pi
- The launcher allows you to access vital parts of your game, and change some core options

While it's been compatible with Raspberry Pi for years, it's never worked as well as it might. On earlier models you needed to tweak some of the memory usage settings as Ren'py could try to use more than was available. Now that Raspberry Pi 4 is here with much more RAM, it works far better – nearly comparable to full-fat desktop machines.

Creating a game

Your humble reviewer spent the last few weeks making a game with it – an item management sim called BBQ DAD (magpi.cc/BBQDAD) – and discovered a huge and helpful community rivalling the one found around Raspberry Pi. Games can easily transcend the intended VN style without too much wrestling, and it has great tools for testing and building your final product. Although you can't quite release to Raspberry Pi just yet. **M**

RUN REN'PY GAMES ON RASPBERRY PI

Ren'py games can be run on Raspberry Pi with some tweaks. After downloading and configuring the launcher for Raspberry Pi, and getting a game, run the Ren'py launcher. Hit Preferences and change the Projects Directory to the folder with the game in. Click Return, select the game from the menu, and hit Launch Project.

10 Best:

Summer projects

Beat the heat with these Raspberry Pi projects

It's getting very sunny here in the UK, so doing electronics projects that involve the great outdoors is fairly safe. Here are some of our favourite summertime projects, some of which might be great all year round. [M](#)



▲ Raspberry Pi GPS Tracker

Relive your journeys

We've seen people use GPS trackers to measure runs, bicycle rides, car rides, and even their high-altitude balloons. The data you get from is a lot more flexible than a smartphone equivalent.

magpi.cc/gpstracker



▲ Raspberry Pi Drone

Fly away

Drones are cool, and it's easier than ever to create one yourself thanks to drone software and light computers... like Raspberry Pi! There are several projects you can follow, with this one using some very specific Raspberry Pi drone tech.

magpi.cc/drone



Alert!

Take great care when working with drones or lasers.

magpi.cc/dronecode
magpi.cc/lasersafety

► Raspberry Pi Roboat

Sail the seas

Upgrading an RC boat to be Raspberry Pi-powered has two advantages: firstly, it's very cool, and secondly, it lets you automate it like a robot. Even if you have to use it in a paddling pool.

magpi.cc/roboat



◀ Stair Lights

What's the temperature?

The stair lights may be indoors, but they tell you what's happening outdoors. How hot is it? Should I put on sun cream or a coat? Before you even head downstairs, you'll be able to tell.

magpi.cc/stairlight



▲ Naturebytes

Wildlife camera

It's a perfect time to set up a wildlife trap, although the Naturebytes camera can be used year round. We've seen people use the tech for simply observing their garden, or to create a bird box for wonderful nest videos.

£110 / \$139 | naturebytes.org

▼ Internet weather station



Simpler meteorology

Don't fancy setting up wind speed and rainfall sensors, barometers, and all that jazz? Simply make it so that you can grab the data from the internet, for a nice, compact little weather update.

magpi.cc/iws

► Automatic chicken feeder

Outdoor animal grub

It may not look much, but this is a very handy way to keep any outdoor pets or animals fed during the summer months. It's easily convertible for water and other kinds of food as well.

magpi.cc/autofeeder

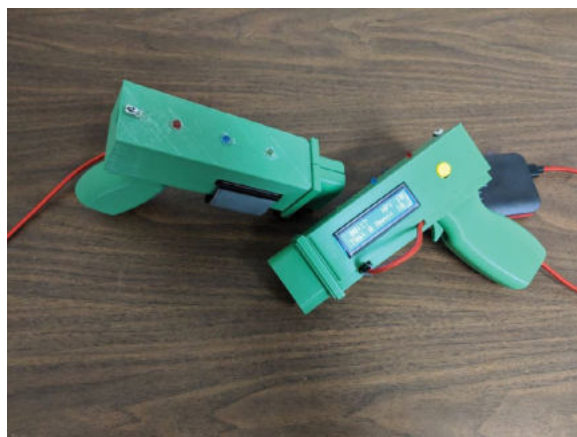


◀ Weather station

Meteorology for all

This build-your-own weather station is an official project that allows you to get professional readings for weather analysis. They can also work together over a small area for more accurate results.

magpi.cc/byows



▲ Infrared Laser Tag

Electronic warfare

A classic form of nineties entertainment re-envisioned for a modern era. We particularly like the very chunky, brightly coloured tag guns. It's almost like a low-polygon aesthetic.

magpi.cc/lasertag



▲ Mini Beast Habitat

Snug as a bug

Need a temporary house for some insects or other tiny beasts? Look no further than this DIY habitat, complete with Raspberry Pi-powered nature camera.

magpi.cc/minibeast

Learn C with Raspberry Pi

Learning how to code in C will also teach you the underlying principles of computing. By **Phil King**

An Introduction to C & GUI Programming

CREATOR

Simon Long


Price:
£10 / \$12

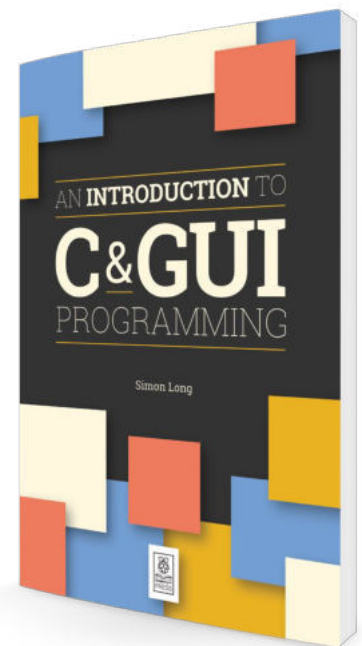
magpi.cc/guibook

Not only does this 156-page official guide offer an introduction to learning C on Raspberry Pi, it shows you how to start creating graphical user interfaces with it.

Aimed at beginners, the first half of the book teaches you the basics of writing simple command-line programs in C. Starting with Hello World, it covers everything from basic variables, strings, and arithmetic, through pointers (which learners often find difficult to grasp), right up to file input and output. Unlike many books on C, which can

run to thousands of pages, you get all the essential information in a compact format with no waffle.

Once you've worked through those chapters (or if you have previous experience using C), you are ready for the second half of the book, which shows how to use the GTK user interface toolkit with C to create feature-rich GUI applications that can be run on the desktop. You start with handling user input with buttons and menus, before moving on to more advanced UI features such as data stores and dialogs. 



Books to read

Study these guides to learn the language

C PROGRAMMING ABSOLUTE BEGINNER'S GUIDE

Ideal for complete beginners, its 352 pages take readers step-by-step through all the concepts with great clarity and numerous code examples. By the end, you'll have a good grasp of the language.

magpi.cc/cpabg

PROGRAMMING IN C

Stephen G Kochan's epic 544-page tome covers everything from the fundamentals to

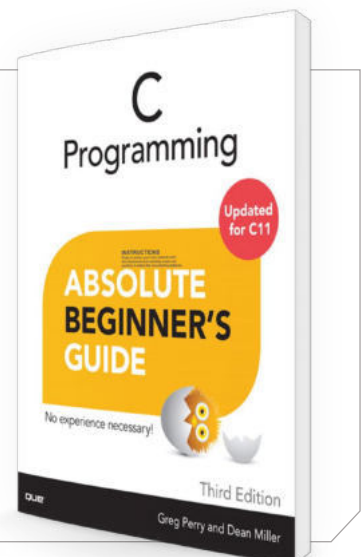
advanced topics in great detail and features practical examples. An ideal companion for an online course.

magpi.cc/ckochan

C PROGRAMMING LANGUAGE

Co-written by Dennis Ritchie, the creator of the C language, this book (aka 'K&R') has long been viewed as the definitive tome for learning C. A little heavy-going, but a trusty reference guide.

magpi.cc/kandr



Learn-C.org

CREATOR

**Ron
Reiter**

Price:
Free

learn-c.org

While not a comprehensive guide to C (it's still under construction), this free set of online tutorials is aimed at complete beginners. As one of a series of Interactive Tutorials sites covering different languages, the vision is to teach

coding within the browser using short and effective exercises. By running actual code directly from the browser – with windows for code and output – learners can try out programming and focus on just that, without having to set up a coding environment locally.

Starting with a simple Hello World example, the Basics section takes learners through topics such as variables, arrays, strings, and functions. The Advanced section moves on to more complex concepts such as pointers, recursion, and binary trees. It should give you a solid grounding in C. learn-c.org



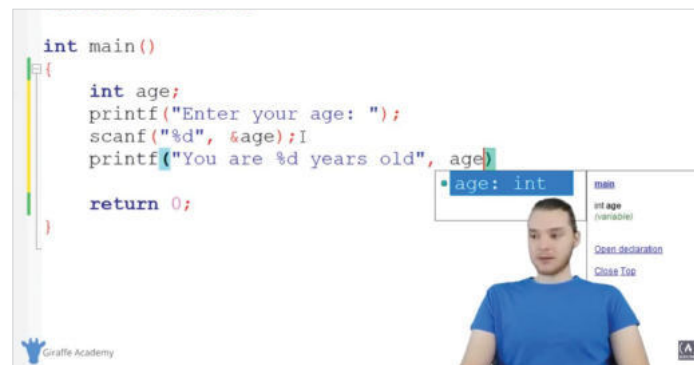
C Programming Tutorial for Beginners

CREATOR

CodeCamp.org

Price:
Free

magpi.cc/codecampc



Learning to code from a video isn't always recommended, but it can complement written and interactive resources. With a running time of nearly four hours, this YouTube tutorial from CodeCamp.org takes you through the basics of the C language. The on-screen narrator has clear diction and explains the concepts in detail, while the enlarged code window appears in the background.

Following an introduction and how to set up everything you need for C on Windows or macOS, you're straight into a Hello World example before moving onto standard topics such as variables, data types, functions, and pointers, along with the creation of a simple calculator and game. A handy list of topics underneath enables you go straight to the section you want. magpi.cc/codecampc

Online courses

Learn C with these interactive courses



CS50'S INTRODUCTION TO COMPUTER SCIENCE

This free massive open online course (MOOC) from Harvard University is a popular starting point for beginners to programming. Most of the content is in C, and there's a cloud-based IDE for coding. magpi.cc/cs50

INTRODUCTION TO PROGRAMMING IN C

Duke University's four-part course aims to develop strong programming fundamentals for learners who want to solve complex problems by writing programs. You'll need to sign up to Coursera (£38 per month). magpi.cc/dukec

ADVANCED C PROGRAMMING: POINTERS

Want a few extra pointers on how to use pointers in C? This focused Udemy course explores the tricky topic in great detail, although it's not aimed at beginners. Code examples are provided. magpi.cc/cpointers



Masafumi Ohta

We talk to Masafumi Ohta about Raspberry Pi and maker communities in East Asia

➤ Name **Masafumi Ohta** | ➤ Occupation **Consultant**
➤ Community role **Community leader** | ➤ Website **raspi.jp**

One of the things we like to do in *The MagPi* is cover the Raspberry Pi community outside of the UK and America. Seeing the different approaches in various countries to making and tinkering is always enlightening, and it's thanks to the efforts of people like Masafumi Ohta that we've been able to see some of the amazing work from makers in Japan and even other parts of East Asia.

"I founded 'Japanese Raspberry Pi Users Group' with [around ten] Raspberry Pi nerds... in October 2012, I always remember the pics we got at Cafe in Jiyugaoka [in Tokyo] to have 'Raspberry Pie' with Raspberry Pi and started our community," Masafumi tells us. "Now we have 150+ active members (600+ subscribed in the Google group), with eight of our 16 committee members being women. We hold meet-ups (Raspberry Jams) mainly in Tokyo and exhibit and present talks at open-source events – Open Source conference, Maker Faire Tokyo/Ogaki/Kyoto."

▶ We've met the Japanese community in person, and they're incredibly creative

▼ The Raspberry Pi User Group can usually be found at Maker Faire Tokyo



What is your background in making and programming?

I graduated from a Japanese technical high school with a major in electric engineering. I have learned not only soldering to make electronic DIY, but also welding and programming with Apple IIe and punch cards.

I also came across the Sharp pocket computer PC-1416G (called Pokecon) as one of the learning materials at that time.

It is for studying BASIC, though. I was trying to analyse its hardware to make some games with its assembly language. I investigated the internal system call to develop a mini graphical game with 'poke, peek, and call' commands, like IchigoJam.

I tore it down, looking into the hardware itself. It was really fun for me to play with it – I am sure that is why I love programming and making.

“ Now I am planning to found a Raspberry Pi business community to help their businesses ”

How did you learn about Raspberry Pi?

Frankly speaking, I was not so interested Raspberry Pi at first because I was still interested in OpenSolaris as a contributor. However, I saw it at an open-source event and tried it, and got interested and thought I would love to try to make some Linux servers with it. Now there are five original Raspberry Pi computers [used for] DLNA, MPD, Shairport (AirPlay), and Icecast streaming. They work really well. I've replaced one of them with Raspberry Pi 3 because of the lack of CPU power, though. Four of the Raspberry Pi boards are almost eight years old and work great.

Supporting a community staying at home

"I am collaborating with the community in many ways: sending some official swag, holding events with them. Lately, almost all of the events are online or cancelled, and I have to find a way to help them [...] – there are many more problems in communicating with others, especially as we cannot meet face-to-face. I hope the situation will get better and we can start growing again our Raspberry Pi community."

What are the maker and Raspberry Pi communities like in Japan?

The Japanese Raspberry Pi community is still growing, but getting commoditised into business-to-consumer (B2C), [but] consumers often change their mind and B2C product life is really short lately.

On the other hand, the Japanese Raspberry Pi B2B business community is now growing rapidly. There are tons of use cases lately: AIoT, prototyping at an automotive company for self-driving, sensors at a vineyard, and more. Now I am planning to found a

Raspberry Pi business community to help their businesses. There are also many more makers lately.

Do you have any involvement in other East Asian communities?

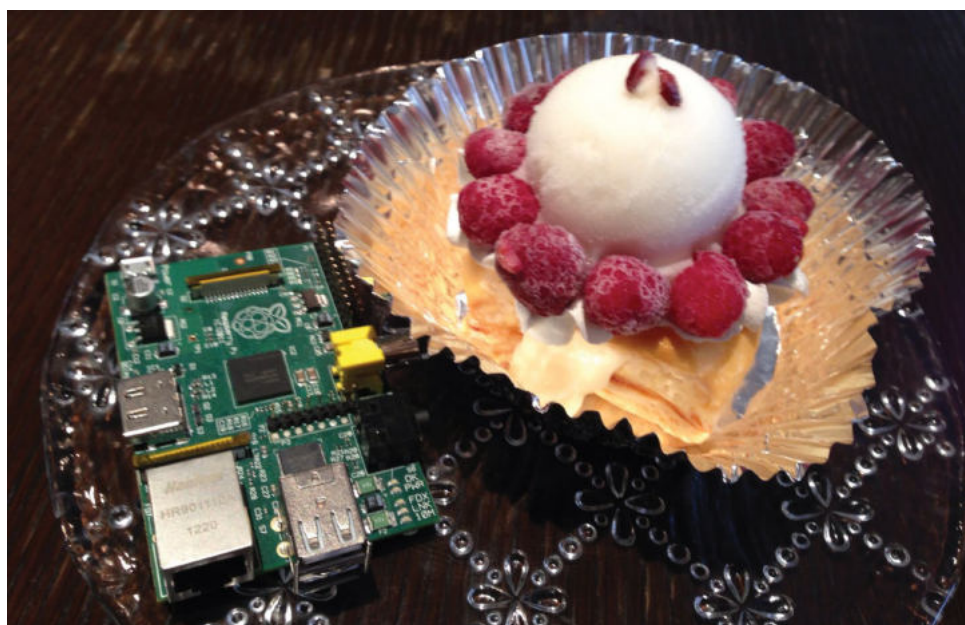
Yes, now I am concerned deeply with East Asian communities, especially helping the Taiwanese. CutiePi is the one of the projects I'm helping spread the word about. I will also help a couple of Taiwanese Raspberry Pi projects this year.

I have been attending COSCUP every year to expand and encourage their community. I am now staying in touch with some Taiwanese IT companies to use Raspberry Pi for their B2B business, and thinking of founding a Raspberry Pi business community there with them. 



▲ Masafumi often presents at Japanese Raspberry Jams and other events

▼ The raspberry pie to celebrate the start of the Raspberry Pi User Group



This Month in Raspberry Pi

MagPi Monday

Amazing projects direct from our Twitter!

Every Monday we ask the question: have you made something with a Raspberry Pi over the weekend? Every Monday, our followers send us amazing photos and videos of the things they've made. Here is a small fraction of them. Follow along at the hashtag #MagPiMonday. [📷](#)

01. This reminds us of futuristic navigation systems from older movies
02. A wholesome felt bee? What is not to love?
03. Here's a nice, fairly quick, and pretty practical project!
04. The teddy head moves (creepily) with Lorraine – it has to be seen to be believed
05. A fairly complex little project that should help with future robotics!
06. No matter how accurate this project is, we'd probably still have to run
07. An interesting use of ultrasonic sensors!
08. More great-looking retro radios upgraded with Raspberry Pi
09. Raspberry Pi-powered soundboards are something we hadn't quite considered, but would be very useful

01

 **John Chinner**  [@JohnChinner](#)

Replying to [@TheMagPi](#)

I have been watching the ISS flying over the earth with my little sense HAT tracker.

 **John Chinner**  [@JohnChinner](#) · Jun 16

I had a spare [@Raspberry_Pi](#) and sense HAT, left over from the glory days of firing two [@astro_pi](#)'s to the space station on a big rocket, so I decided to make the "World's Smallest Real Time ISS Ground Track Map". But how, I hear literally none of you say?

1/n
(*Needs citation)
[Show this thread](#)



8 Bits and a Byte
@8BitsandByte

Replying to @TheMagPi

An internet controlled bumble bee, sharing fun facts and supportive statements!



0:03 849 views

Richard Hayler
@rdhayler

Replying to @TheMagPi

I've been enjoying playing darts in the garden recently but it was too windy this weekend. But the chalkboard I normally use for scores creates too much dust indoors so I made a silly score-keeper using a @Raspberry_Pi Zero



Scott Young
@thebot75

Replying to @TheMagPi

Now that the buses are running up here again built this bus tracker for the stop outside my house. LCD backlight has a green / yellow / red status for how long it is until the next bus (if it's red you'd better start running 😊). #MagPiMonday



0:09 842 views

Lorraine Underwood
@LMCUnderwood


Replying to @TheMagPi

I was trying out a thing!

Lorraine Underwood @LMCUnderwood · Jun 27

Note to self: a shreddies box is no good as a tripod 🤖

[Show this thread](#)



0:22 2.2K views

Dr Footleg - Roboteer
@drfootleg

Replying to @TheMagPi

How did I miss #MagPiMonday?! I even had project progress to report. My first Arduino-Pi integrated build and a load of learning to help me develop software and documentation to support my robotics controller board hardware project.

Dr Footleg - Roboteer @drfootleg · Jul 2

Solid progress on organising the code to support my #Sentinel robot controller board this evening. Refreshed my knowledge of setting up virtualenvs, reorganised the code in to a packaged library and deployed in develop mode in a virtualenv. Confirmed by importing it into python.


[Show this thread](#)



Olog-Hai
@Moonglum

Replying to @TheMagPi

Horsed around with Python and made a matrix keyboard soundboard.



Jitesh Saini
@jiteshsaini10

Replying to @TheMagPi

Smart Water Tank project: Using #RaspberryPi and ultrasonic distance sensor. Created an online dashboard to monitor real-time level of water in my overhead tank. Check out this link to see it yourself helloworld.co.in/water-tank #IoT

IOT Project: Smart Water Tank



0:08

Sickly_b
@sickly_b

Replying to @TheMagPi


Finished this last week

Sickly_b @sickly_b · Jul 2

I finished my Roberts Rambler - it's now a Stereo Networked Music Player with rechargeable power and USB charge and flash drive ports, running @volumio on a @Raspberry_Pi 3A+

Parts from @pimoroni and @ThePiHut 🙌

[Show this thread](#)



1:57 2.8K views

Raspberry Pi 4 for educators

George Silberstern shows us how he's helping his teacher partner with distance learning

There's a lot of online and distance learning happening right now. While there are plenty of reports about teachers and educational bodies adapting to it, there are untold stories of teachers for whom the switch may not be so straightforward.

George Silberstern emailed us recently about his partner's effort for distance learning: "She used to handle all of her work-related, non-teaching duties on her classroom PC, even though the related applications are web-based and remotely accessible. That was partially because she didn't own a computer and couldn't justify the cost of a basic, functional laptop when she could keep work at work."



▲ George's partner teaches SEND students in an American elementary school

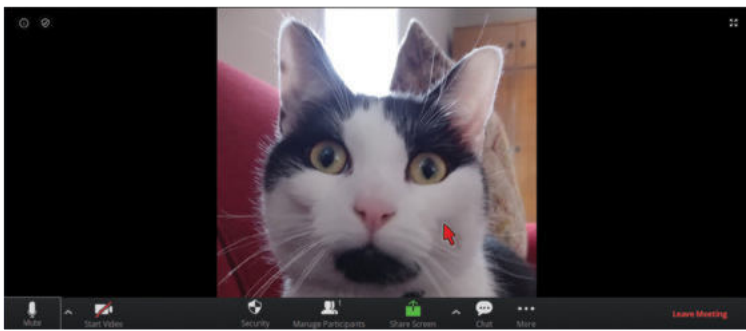
When teaching moved online, George decided to experiment with a Raspberry Pi 4 and Ubuntu, although he had to get creative.

"The most current release of Ubuntu MATE for Raspberry Pi is 18.04 LTS, whereas the Ubuntu world stands at 20.04 LTS. Ordinarily this wouldn't be a big deal. However, support for underlying releases expiring well before a Raspberry Pi build saw an update to the next release has happened

“ He decided to experiment with a Raspberry Pi 4 and Ubuntu ”

before [...]. Moreover, the 18.04 release will not run on Raspberry Pi 4! Not all is lost, however: Ubuntu MATE is nothing more than a customisation of Ubuntu Linux bundled with the MATE Desktop. Since the stock Ubuntu release does run on Raspberry Pi 4, the same result can be achieved the old-fashioned way. For my partner's installation, I used the 64-bit Ubuntu Server 19.10 image."

While there are still some issues due to some of the sites the curriculum requires, the difference is night and day once properly configured. ■



▲ Attempts with Zoom were made

▶ The new Raspberry Pi 4 build after a less-than-optimal Raspberry Pi 3B+ attempt



Crowdfund **this!** Raspberry Pi projects you can crowdfund this month



CutiePi Tablet

A nice little tablet concept that we've featured in the magazine before. CutiePi is finally up on Kickstarter, and has been smashing all its goals and stretch goals, so it might be a good time to hop on.

► kck.st/3dGXoks



CrowPi2

We just missed this as we went to press last month – CrowPi2 is a laptop modification kit for Raspberry Pi that also has a lot of STEM education functions. It reminds us a little of the Piper laptop build. It's blown way past its target as well, so give it a look.

► kck.st/3ftWitM

CROWDFUNDING A PROJECT?

If you've launched a Raspberry Pi-related project, let us know!
magpi@raspberrypi.org

Best of the rest! Here are some other great things we saw this month

MINECRAFT FRIEND NOTIFIER

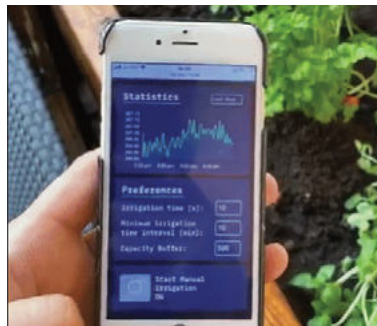
A great-looking piece that lights up as more and more of your friends join your server.



► magpi.cc/aXvpCv

AUTOMATED IRRIGATION SYSTEM

This Raspberry Pi-controlled irrigation system with a custom web app is a great mix of technologies.



► magpi.cc/bTxrkH

SUBMERGED RASPBERRY PI

Putting computer parts in oil is a cooling technique for big computing rigs. Or in this case, one part cooling solution, one part great little display.



► magpi.cc/GApimB

POSSIBLE EVENT CANCELLATIONS

Please follow local public health advice and take decisions on whether to cancel clubs or events in consultation with the venues that host them.

Raspberry Jam Event Calendar

Find out what community-organised Raspberry Pi-themed events are happening near you...

01. Build a web server using Raspberry Pi

📅 Saturday 1 August

📍 Online

▶ magpi.cc/e6zvqf

Learn the basics of the web, the structure of web programming, and how to deploy a simple local web server.

02. Video Game Developer Bootcamp

📅 Monday 3 August

📍 Online

▶ magpi.cc/a3rqVS

Aimed at 11- to 13-year-olds, this camp helps you build your own handheld video game console and develop games with code.

03. Virtual CoderDojo

📅 Tuesday 4 August

📍 Online

▶ magpi.cc/69EQGp

Attendees are welcomed at 1:30 pm and will experience virtual 'stations' for online coding activities.

04. Houston Raspberry Pi Meetup

📅 Wednesday 5 August

📍 Online

▶ magpi.cc/W2Ugxx

A community of coders, artists, educators, and engineers interested in project-based learning with Raspberry Pi.

05. Girls Into Coding

📅 Saturday 8 August

📍 Online

▶ magpi.cc/iEgRLi

Have fun participating in exciting digital and coding workshops, listening to inspiring talks, and controlling robots from your home!

06. The Repair Café Experiment

📅 Saturday 8 August

📍 Online

▶ magpi.cc/5g7c2w

Find out how to repair, mend, make, hack, or repurpose your things at home online.

07. Online Summer Camp: Python Coding

📅 Monday 10 August

📍 Online

▶ magpi.cc/f8ujRW

Through this introductory coding course, your child will gain familiarity with the Python programming language.

08. Medway Makers Tech Meetup

📅 Sunday 16 August

📍 Online

▶ magpi.cc/qAyyz6

Workshops in building cool things with Arduino, how to code for beginners, how to use a Raspberry Pi, and so on.

FULL CALENDAR

Get a full list of upcoming events for August and beyond here:

rpf.io/jam



FIND OUT ABOUT JAMS

Want a Raspberry Jam in your area? Want to start one?

jam@raspberrypi.org



We've highlighted some of the areas in need of a Jam! Can you help out?



Raspberry Jam advice:

Start a mailing list

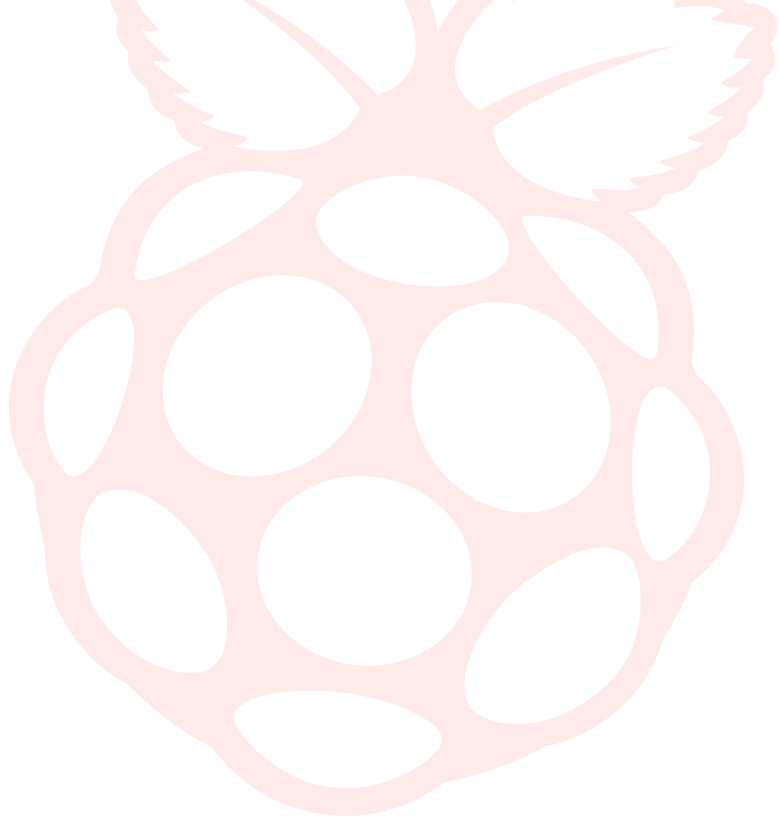
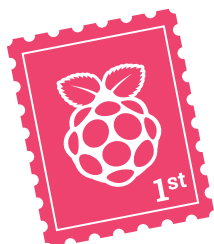
"Having a staff mailing list for all your active volunteers really helps: mailmanlists.net run email discussion lists, and they give free accounts to worthy causes like Raspberry Jams."

Andrew Oakley – Cotswold Jam

Every Raspberry Jam is entitled to apply for a Jam starter kit, which includes magazine issues, printed worksheets, stickers, flyers, and more. Get the book here: rpf.io/guidebook



Your Letters



▼ Looking for shots this close up? You might want to check out the microscope lens

Water-cooled

Just recently gone through an upgrade process for my computer and it got me thinking – is there any reason to water-cool my Raspberry Pi 4? I know they can get a bit hot, but is it worth it? Are there even kits for it?

Henry via Facebook

We've not come across water-cooling kits ourselves for Raspberry Pi, but even if you're really trying to push it past its limits, you wouldn't need one. Heat output on Raspberry Pi 4 doesn't interfere with performance under regular use, and even if you're overclocking, a case with cooling (e.g., a heatsink or fan) is good enough.

▼ Cases with aluminium, heat-dissipating designs help to regulate temperature on Raspberry Pi



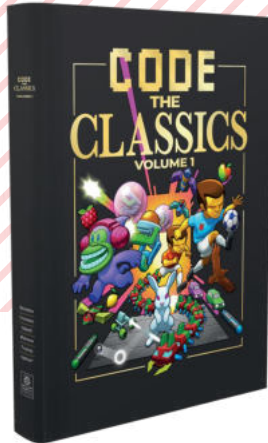
Macro shots

I really like the new Raspberry Pi High Quality Camera – it's got me a bit inspired to do some fun photography projects, including some time-lapse stuff of my flowers.

I think it would be really cool if I could do some excellent macro shots, though. Is there a lens I can use for that?

Stephen via email

If the standard 6mm lens isn't enough for you, Pimoroni has just released a microscope lens (magpi.cc/microscopelens) that can create some fantastic, incredibly close-up shots. So if you want to see some tiny bugs or individual pollen particles, this might be the one for you.



Making games

My daughter is using a Raspberry Pi to learn coding and has taken a bit of a shine to Python. She wants to try to make games like she does in Scratch, but seems to be a bit lost in where to start. Do you have any tips for this? Is there anything like Scratch for Python for making games?

Ann via email

Unfortunately, there's nothing in Python really equivalent to Scratch for making a game out of building blocks and assigning them to sprites. However, Python is great for learning how to program games and such – a specific gaming library called Pygame (pygame.org) is often used for this purpose and the Pygame Zero wrapper for it (magpi.cc/pygamezero) makes it even easier and more intuitive. The recent *Code the Classics* book (magpi.cc/codetheclassics) has a load of classic games recreated in Pygame Zero to learn from.

We also recently had a go with Ren'py, a visual novel engine that works on Python. You can read our full review of it on page 79. However, we think it's pretty good for learning about more story-driven games, and it has a great community behind it.



▲ Some keyboards are a bit more universal than others, but if you're used to one layout it becomes second nature

More keyboards

I've been wondering if there will ever be any more keyboards in other languages offered by Raspberry Pi? Have you heard anything at all?

Tuisku via Twitter

You're in luck! Raspberry Pi announced a number of new keyboard types around the time we were making this issue of the magazine. Depending on where you live, you may be happy to know that you can now get classic layouts that you'd find in Portugal, Norway, Sweden, and Denmark.

Still missing a keyboard type you'd prefer? Make sure to let the folks at Raspberry Pi know, otherwise they may not know to look into it.

Contact us!

- Twitter [@TheMagPi](https://twitter.com/TheMagPi)
- Facebook [magpi.cc/facebook](https://facebook.com/magpi.cc/facebook)
- Email magpi@raspberrypi.com
- Online raspberrypi.org/forums

remot³.it

Can't port forward?

No global IP address? No admin access to router settings?

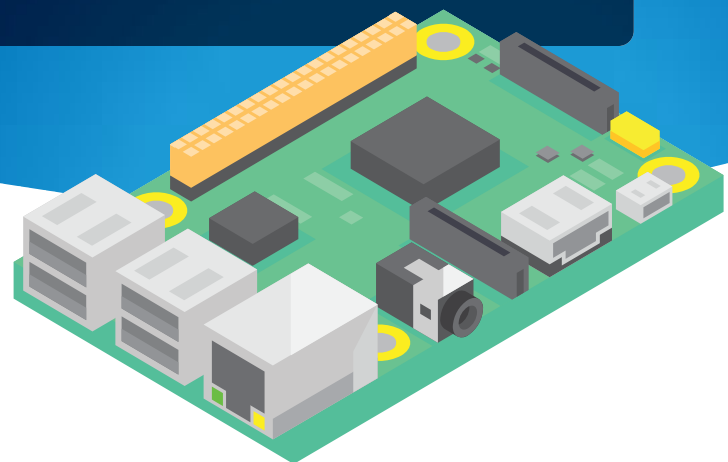
On a mobile network? Device behind multiple subnets? Worried about leaving open ports? Worried about exposing your global IP?

No Problem.

Use your localhost address to create secure tunnels to your remote devices.

Install in an instant on Raspberry Pi OS

```
> sudo apt update  
> sudo apt install remoteit
```



Port forwardless access to remote devices

Easily access your Raspberry Pi from anywhere without a public IP address or port forwarding. Supports SSH, VNC, web servers, Minecraft, and more. Free for personal use. Business licenses available. remote.it is an approved package for Raspberry Pi OS. Install in moments!

Download remote.it at
remote.it/rpi



WIN ONE OF TEN ARGON NEO CASES AND FAN HATS

The newest case from the makers of the Argon ONE is much sleeker and comes with amazing active cooling. There are ten up for grabs.



Head here to enter: magpi.cc/win | Learn more: argon40.com

Terms & Conditions

Competition opens on **29 July 2020** and closes on **27 August 2020**. Prize is offered to participants worldwide aged 13 or over, except employees of the Raspberry Pi Foundation, the prize supplier, their families, or friends. Winners will be notified by email no more than 30 days after the competition closes. By entering the competition, the winner consents to any publicity generated from the competition, in print and online. Participants agree to receive occasional newsletters from The MagPi magazine. We don't like spam: participants' details will remain strictly confidential and won't be shared with third parties. Prizes are non-negotiable and no cash alternative will be offered. Winners will be contacted by email to arrange delivery. Any winners who have not responded 60 days after the initial email is sent will have their prize revoked. This promotion is in no way sponsored, endorsed or administered by, or associated with, Instagram or Facebook.

Wireframe

Join us as we lift the lid
on video games



Visit wfmag.cc to learn more

LOW-COST

3D Printing

& Making

**DIY 3D PRINTING, LASER
CUTTING, AND CRAFTING
WITH RASPBERRY PI**

**THE MAGPI #97
ON SALE 27 AUGUST**

Plus!

Build a software
preservation station

**Make your own robot
from scratch**

Hack a retro
television

DON'T MISS OUT!

magpi.cc/subscribe

TWITTER @TheMagPi

FACEBOOK fb.com/MagPiMagazine

EMAIL magpi@raspberrypi.com

EDITORIAL

Editor

Lucy Hattersley
lucy@raspberrypi.com

Features Editor

Rob Zwetsloot
rob.zwetsloot@raspberrypi.com

Sub Editors

Phil King and Nicola King

ADVERTISING

Charlotte Milligan
charlotte.milligan@raspberrypi.com
+44 (0)7725 368887

DESIGN

criticalmedia.co.uk

Head of Design

Lee Allen

Designers

Sam Ribbits, Harriet Knight

Illustrator

Sam Alder

CONTRIBUTORS

Mike Cook, David Crookes,
PJ Evans, Rosemary Hattersley,
Nicola King, Rob Miles,
KG Orphanides

PUBLISHING

Publishing Director

Russell Barnes
russell@raspberrypi.com

Director of Communications

Liz Upton

CEO

Eben Upton

DISTRIBUTION

Seymour Distribution Ltd
2 East Poultry Ave,
London EC1A 9PT
+44 (0)207 429 4000

SUBSCRIPTIONS

Unit 6 The Enterprise Centre
Kelvin Lane, Manor Royal,
Crawley, West Sussex, RH10 9PE
+44 (0)1293 312193
magpi.cc/subscribe
magpi@subscriptionhelpline.co.uk



This magazine is printed on paper sourced from sustainable forests and the printer operates an environmental management system which has been assessed as conforming to ISO 14001.

The MagPi magazine is published by Raspberry Pi (Trading) Ltd., Maurice Wilkes Building, St John's Innovation Park, Cowley Road, Cambridge, CB4 0DS. The publisher, editor, and contributors accept no responsibility in respect of any omissions or errors relating to goods, products, or services referred to or advertised in the magazine. Except where otherwise noted, content in this magazine is licensed under



a Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Unported (CC BY-NC-SA 3.0).
ISSN: 2051-9982.



A nice little earner

How coding can help us build a smarter society

As I'm writing this, Twitter has been hacked and lots of official blue-tick channels are busy promoting the doubling of Bitcoin investments. Fortunately, I don't think any Raspberry Pi accounts were involved.

I hope nobody who reads this magazine fell for the scam. We like to think you're all a pretty savvy bunch.

But sympathy if you did. I once did an honesty session with a bunch of technology experts and was surprised at how many had fallen for a phishing scam at least once.

“We are proud to help as many people as possible learn to code”

I really hope nobody who reads *The MagPi* was involved in the scam itself. If you were, I say, “boo to you!” Go read some other publication instead.

Besides. If you want to make money with computing, then scamming isn't the way to do it. Learn computing and coding; learn to control computers. It teaches you to think, to automate boring tasks, and opens up a world of creative fun.

Let's create

With coding under your belt, you can make games, apps, play around with electronics, build incredible widgets

and gizmos. All of which is much more fun than hustling for cyber-currency on Twitter.

This month we wrote a feature on no-nonsense projects. While this may seem trivial to some of our long-term readers, this sort of thing is vital to newcomers. We are keenly aware of our role in helping newcomers overcome the initial hurdles and learn to love computing.

The real, very serious, purpose of Raspberry Pi is to “put the power of computing and digital making into the hands of people”. We think that

once we help people understand computers, they can use them to solve the problems that matter to them and “express themselves creatively”. Speaking of which, have you seen Rob's new game, BBQ DAD (magpi.cc/bbqdad)?

We live in a world of increasing digital complexity. A world where technologies like machine learning and algorithms might as well be ‘magic’ and ‘witchcraft’ to much of the population.

The best thing you can do – the very best thing for both yourself and the world in general – is to learn

how computers work. Make sure you grok the stuff that's going on here. Not only will you be less likely to fall for scams (whether monetary, psychological, or political), you'll be more likely to end up a more rounded person with a secure life.

Teach a person to fish and they'll eat for a lifetime. Teach a person to program and you'll frustrate them for a lifetime. As the joke goes.

Frustrating though coding may be, it's a lovely skill that's well worth knowing. You may not get mega-rich, but you're more likely to earn the kind of money where you don't need to be tempted by get-rich-quick schemes. And even if you don't, you'll be able to create and make something that you're proud of.

Computing is a ‘nice little earner’, as the TV show *Minder* once taught us. It's also one of the best ways to level up the parts of our country being hit by hardship. I speak from experience here. Computing has helped me move onwards and upwards in life.

We are proud to help as many people as possible learn to code. **”**

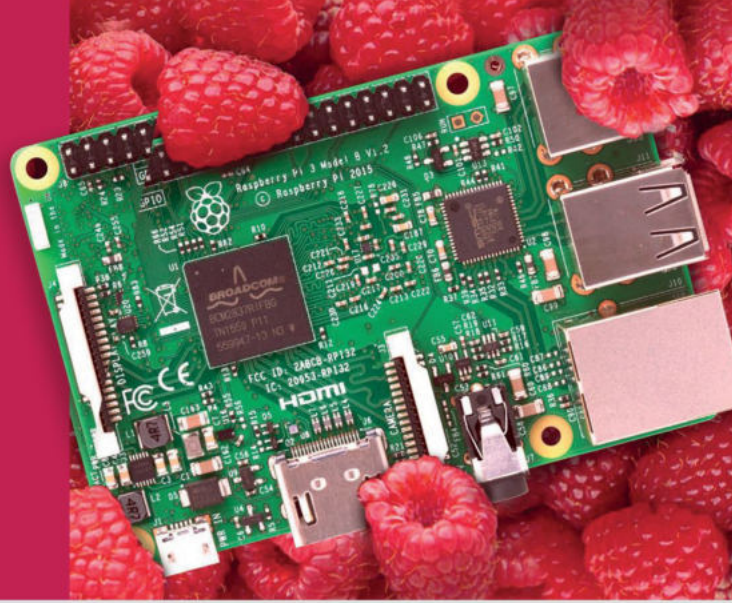
Lucy Hattersley

Lucy is editor of *The MagPi* and is currently playing around with GUIs and Nand2Tetris. When others go low, she goes low and high at the same time.

magpi.cc

AUTHOR

American Raspberry Pi Shop



- Displays
- Cases
- Project Kits
- Add-on Boards
- HATs
- Arcade
- Cameras
- Cables and Connectors
- Sensors
- Swag
- Power Options
- GPIO and Prototyping

Partner and official reseller for top Pi brands:



and many
others!

Price, service, design,
and logistics support for
VOLUME PROJECTS

USA

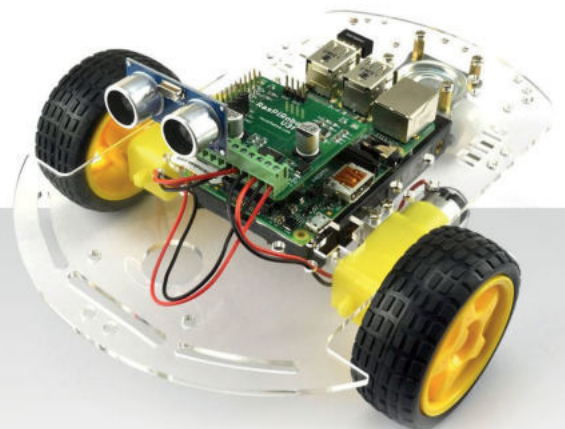


PiShop.us

Canada



BuyaPi.ca



Raspberry Pi

APPROVED RESELLER

RASPBERRY PI 4 STARTER KIT



This 8-piece kit includes everything you need to get going in the world of Raspberry Pi— from all the hardware in a single box, to access to an online step-by-step guide as well as projects to get up and running faster than ever



**Find out more about this
and many other products at**

www.okdo.com

